



23.6.26-AI7기

지난시간 복습 (fashion MNIST로) (~ 40분까지 풀어주세요.)

MLP vs CNN

MLP (multi-layer perceptron)

CNN(convolutional neural network) ⇒ 이미지분석 원통!

Image * filter = convolved feature 연산과정

padding ⇒ 가장자리 정보를 유지하기 위해

stride (default= 1) ⇒ 필터를 옮기는 거리

pooling ⇒ max pooling / average pooling

RGB (red green blue)

MLP vs CNN 차이점

Class(클래스) / instance(객체)

CNN 모델들 (역사적으로만)

vgg 16

Resnet

기타 CNN 모델들

추가자료

기본 이미지 데이터들

퀴즈 (10분)

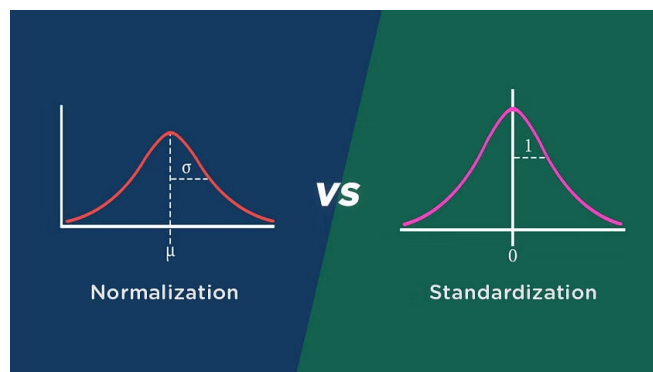
지난시간 복습 (fashion MNIST로) (~ 40분까지 풀어주세요.)

1. 흑백 or 컬러이미지 데이터는 픽셀로 이뤄져 있으며 0에서 (255) 사이의 값을 갖는다.

1-1. 이미지데이터 픽셀을 정규화하기 위해 어떤 방법을 사용하는가? 1로 나누기

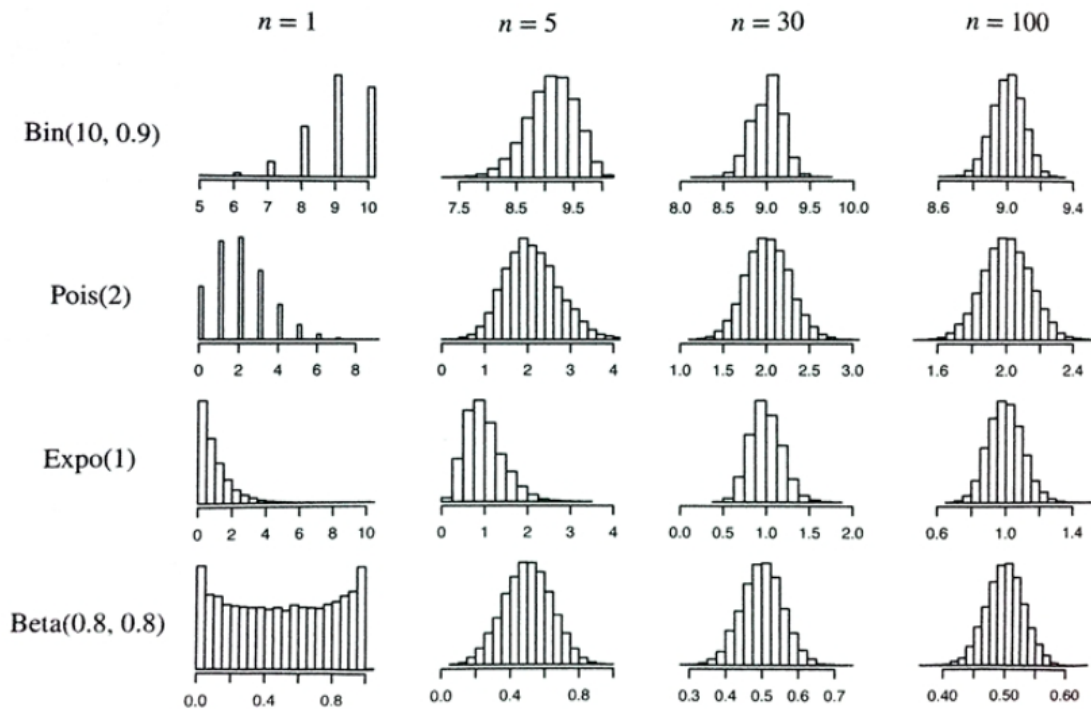
▼ 참고자료

정규화 종모양 ⇒ 왜 'normalize' 시키지?



central limit theorem(중심극한정리) : 표본의 수가 무한히 크다면 확률분포가 정규분포를 이룬다.

아래 그림은 각 확률분포의 표본이 많아질수록, 표본평균은 정규분포에 근사하는 모습을 보여줍니다.



- explainable AI (EX) ⇒ **설명가능한** 인공지능 / 지속가능한 환경보호
- 인공지능과 통계 (확률, 분포)

표본/모집단, 표준편차, 조건부확률, p-value (0.05), T검정, 1종오류, 2종오류, 베이즈 정리, 마르코프 몬테카를로..

AI 연구자를 위한 통계적 학습론

http://www.kmooc.kr/courses/course-v1:PNUk+SL_C01+2020_KM_015/about

2. 전처리 단계에서 train / test 데이터로 나눌때 기본적인 split 비율은 얼마인가? 8: 2

2-1. train 데이터를 다시 한번 test 데이터 용도로 쪼개는 방식이며, 데이터양이 적을때 교차검증용도로 사용할 수 있는 데이터셋은? validation_set

3. 패션MNIST 데이터로 모델 학습 시, 아래 빈칸을 채우시오

```
tf.keras.layers.Dense(10, activation='softmax') # 10, softmax

model.compile(loss='sparse_categorical_crossentropy',
              optimizer='adam',
              metrics=['accuracy'])
```

4. 데이터 칼럼을 왼쪽에서 오른쪽과 같이 변환하는 방법은? 원핫인코딩 :

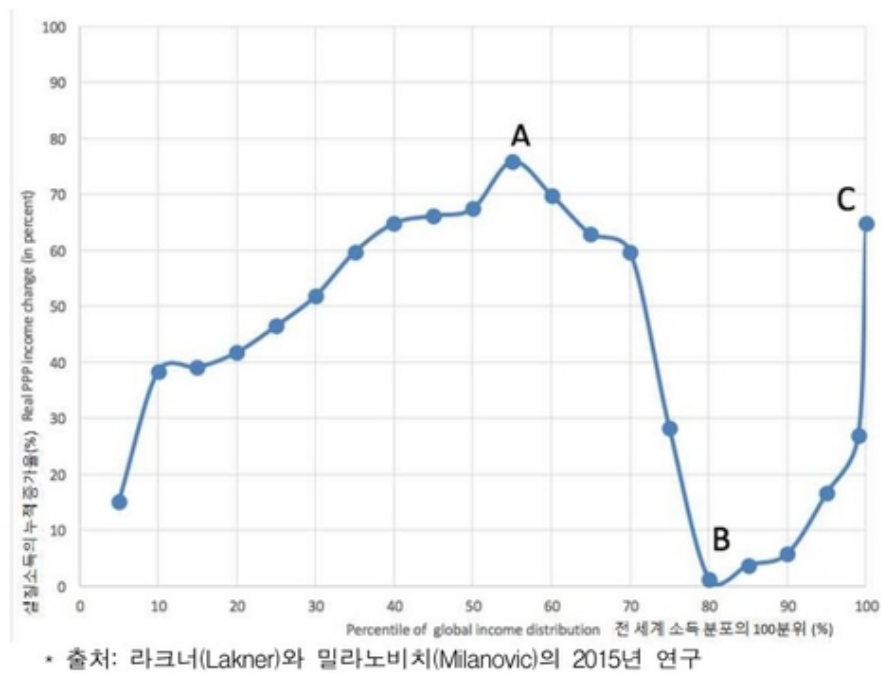
왜? 수우미양가 (A학점, B학점, C학점 ,,) / 색깔, 동물,,, ⇒ 원핫인코딩 (df.get_dummies)

COLOR
red

COLOR_red	COLOR_blue	COLOR_green
1	0	0

red	1	0	0
blue	0	1	0
green	0	0	1
blue	0	1	0
red	1	0	0

5. SGD(stochastic gradient descent)에서 learning_rate와 같이 인간이 설정할 수 있는 인자를 뭐라고 하는가? 하이퍼 파라미터 (hyper parameter)
6. 아래 그래프를 loss 그래프라고 가정했을때, 우리가 찾아야하는 지점은 무엇인가? 알파벳으로 답하라. B (global minimum : 전체적인 최솟값)



7. 다음 설명을 읽고, 무엇에 대한 내용인지 써라. 과적합(과대적합, overfitting)



1. 기계학습이 해결해야 할 최대과제 중 하나.
2. 이 모형은 새로운 데이터에 대해 제대로 학습 및 예측할 수 없으므로 일반화가 불가능하다.
3. 훈련 데이터가 충분하지 않을때 주로 발생
4. 이를 해결하기 위한 방법으로 early stopping, 가지치기, DropOut, 정규화, L1규제, L2규제, 앙상블, 데이터 증강, 전이학습 등의 방법을 사용

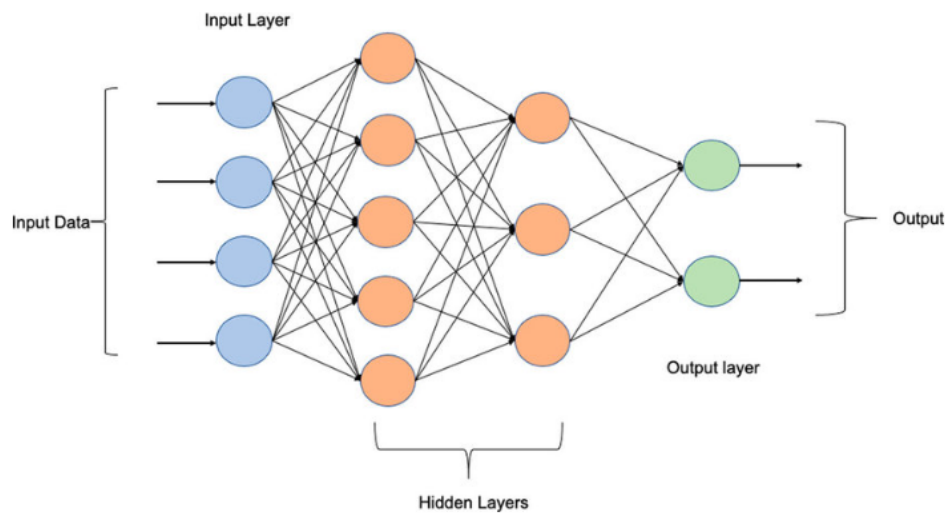
▼ 참고자료

https://yngie-c.github.io/machine_learning/2020/04/30/training_test_reg/

MLP vs CNN

- 퍼셉트론 vs NN(신경망) = 마지막 활성화함수의 유무

MLP (multi-layer perceptron)



CNN(convolutional neural network) ⇒ 이미지분석 원툴!

- feature(특징)을 추출한다. ⇒ filter (kernel)

Convolutional Neural Network

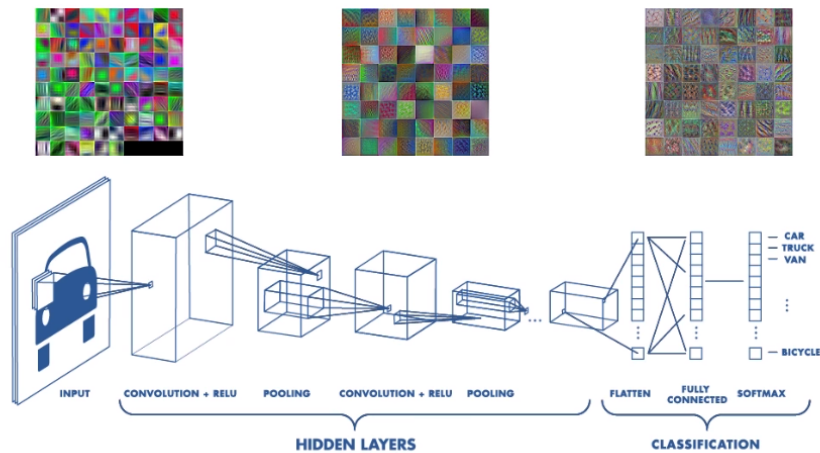
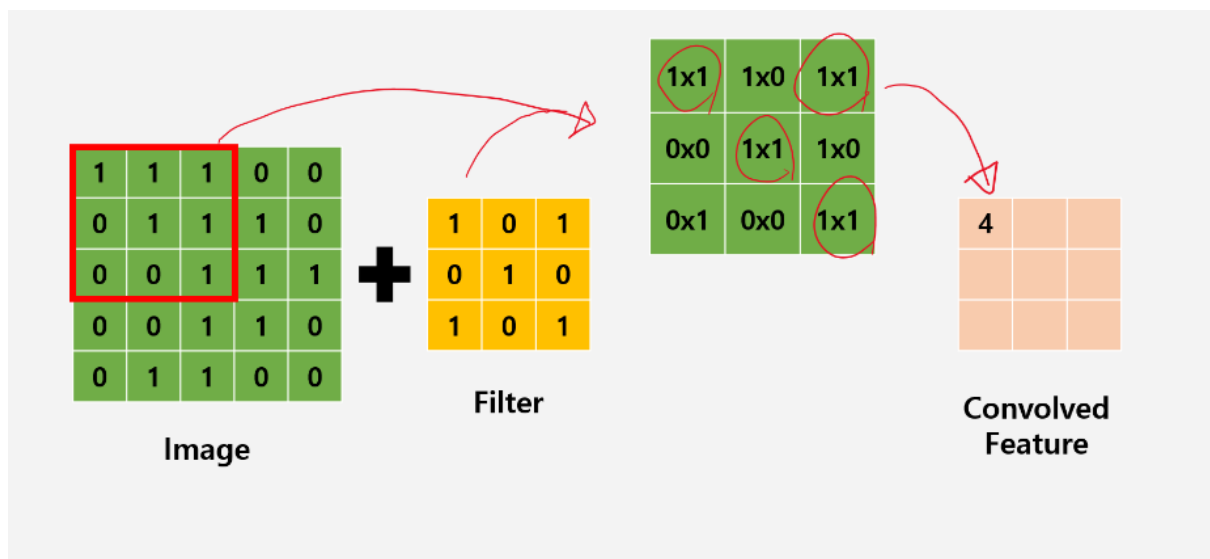
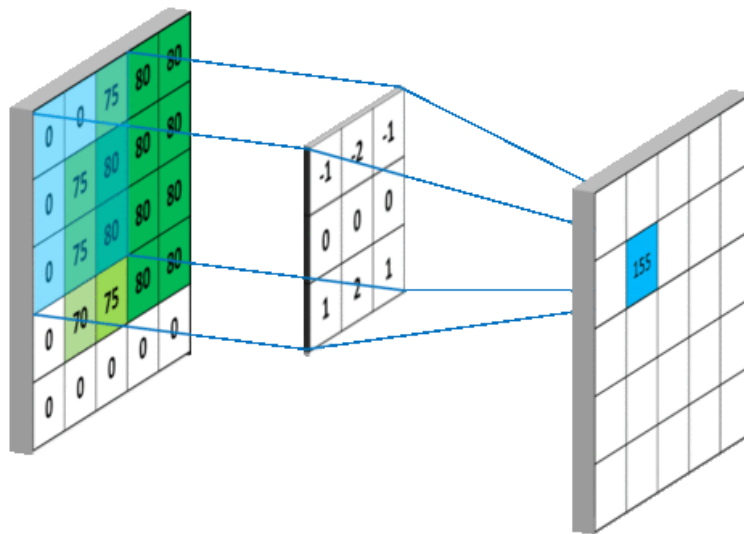
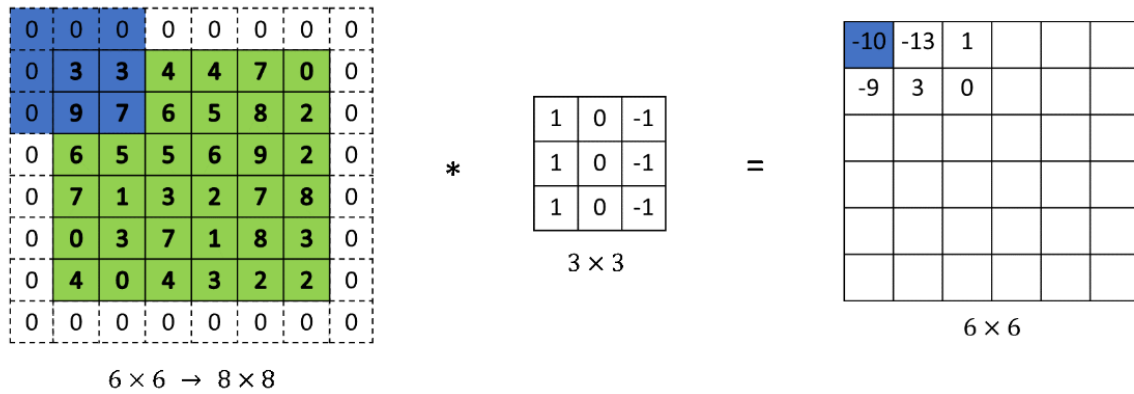


Image * filter = convolved feature 연산과정

- inner product (내적, dot product) = $x_1 * k_1 + x_2 * k_2 + x_3 * k_3 + \dots$ = convolved feature

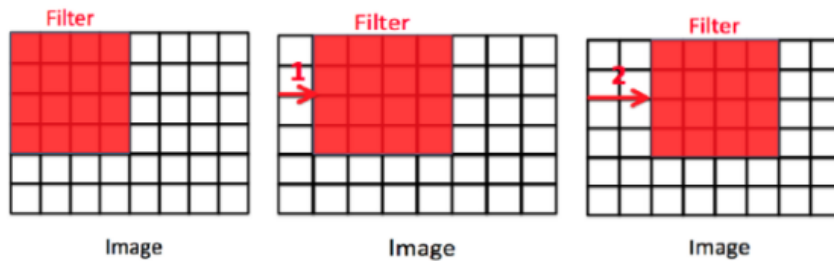


padding ⇒ 가장자리 정보를 유지하기 위해



stride (default= 1) ⇒ 필터를 옮기는 거리

- if 스트라이드가 커지면 convolved feature 크기가 작아진다.



pooling ⇒ max pooling / average pooling

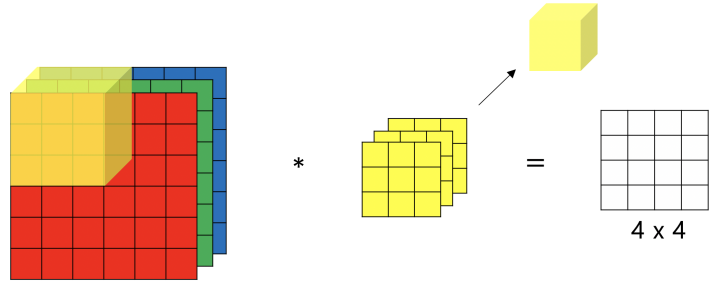
- MAX POOLING 특징

: 소수 parameter 사용하므로 과적합 방지

: 연산이 빠름



RGB (red green blue)



0	0	0	0	0	0	...
0	156	155	156	158	158	...
0	153	154	157	159	159	...
0	149	151	155	158	159	...
0	146	146	149	153	158	...
0	145	148	148	148	158	...
...

Input Channel #1 (Red)

-1	-1	1
0	1	-1
0	1	1

Kernel Channel #1

↓

158

+

1	0	0
1	-1	-1
1	0	-1

Kernel Channel #2

↓

-14

+

0	1	1
0	1	0
1	-1	1

Kernel Channel #3

↓

653

+ 1 = 798

↑

Bias = 1

-25	466	466	475	...
295	787	798
				...
				...
...

Output

MLP vs CNN 차이점

[딥러닝] 컨볼루션 뉴럴 네트워크 Part. 1 - MLP의 문제점과 CNN

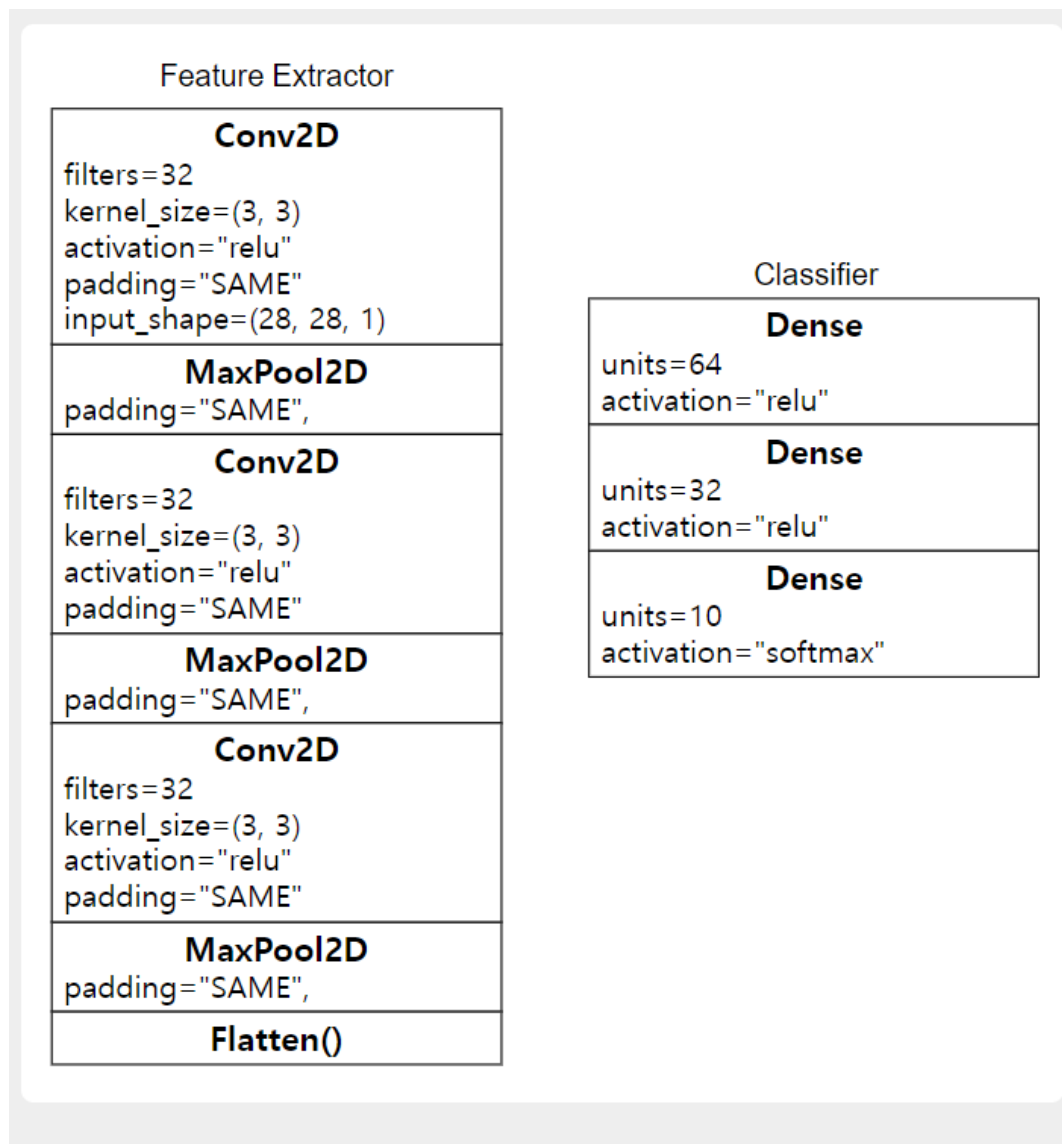
이전까지의 내용 이전 글에서는 각 층의 출력값 자체를 안정화하는 방법인 배치 정규화를 다루고 학습식을 개선할 필요성에 대해 살펴보았습니다. 이번 글에서는 학습식을 어떻게 개선해나가고 있는지에 대해 살펴보겠습니다. 전체 내용은 SlideShare 에서 한번에 보실 수 있으며 youtube 로도 볼 수 있

<https://gomguard.tistory.com/188>

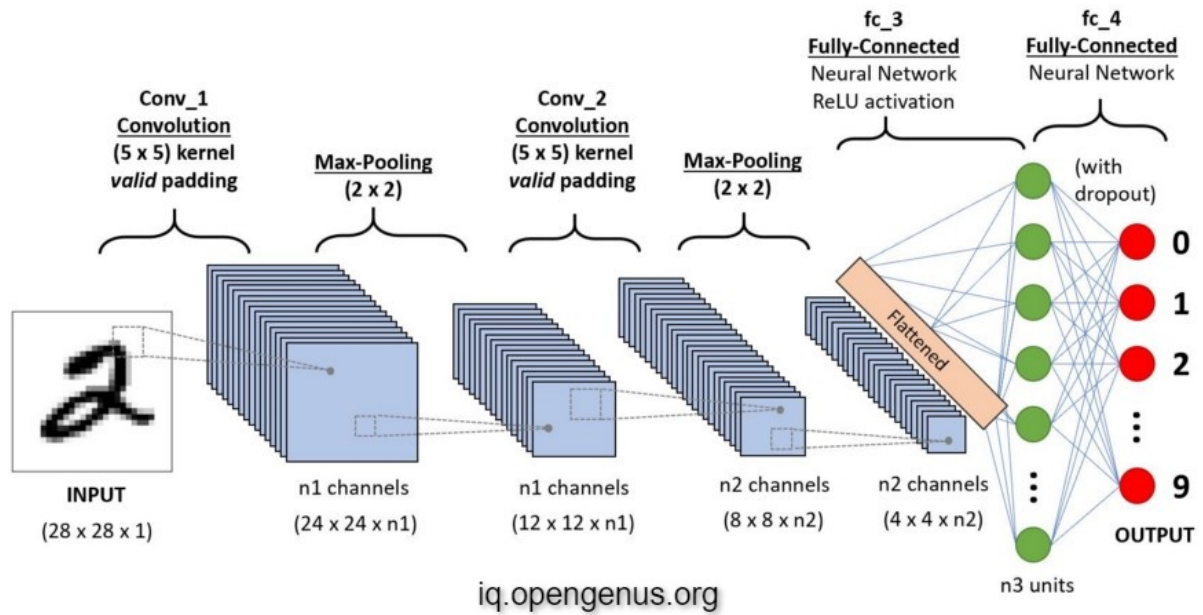


ML
Convolution Neural Network

- filter의 유무 - 특징점 추출
- MLP는 flatten 후, 가중치를 계산하므로 정보손실이 크다
- CNN 2차원 행렬형태로 가중치 계산 후, flatten한다.(2차원 행렬 \Rightarrow 정보손실이 적고, 빠름)



- FC layer == MLP == Affine layer



- MLP \Rightarrow 1차원
- CNN \Rightarrow 2차원

Class(클래스) / instance(객체)

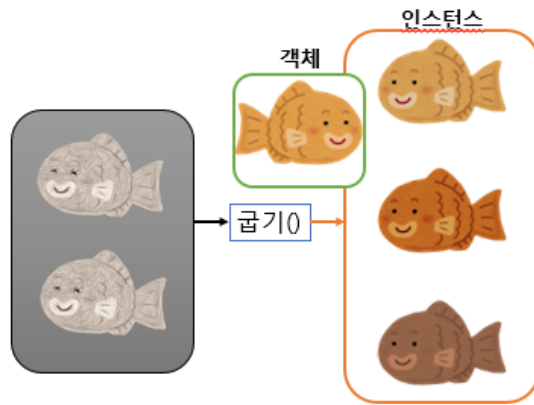
```
def multi(num1,num2) :
    return num1 **2
```

```
multi(1,3) =>
```

자주사용하는 함수들은 클래스에 묶어놓기.

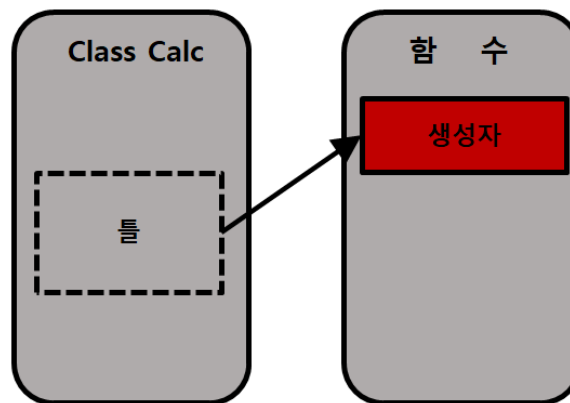
```
def 1
def 2
def 3
def 4
def 5
def 6
def 7
```

```
=> class 1~7
```

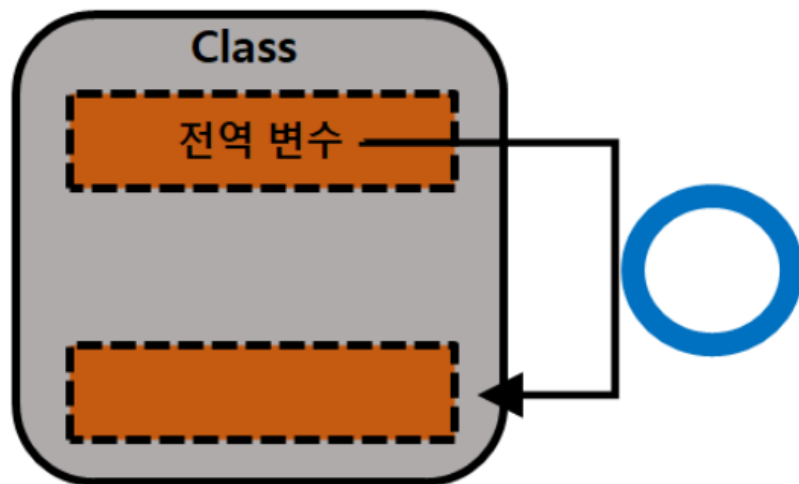
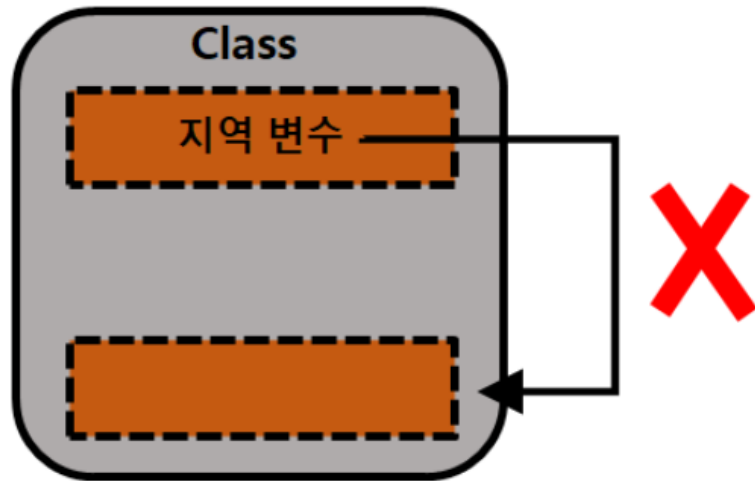


- 클래스: 붕어빵틀/ 객체: 붕어빵 한개

- `__init__` (생성자) : 클래스 인스턴스 초기화. 클래스 작성시 필수



- init 메소드의 첫번째 인자는 self로 지정! (`self.변수`)
- `self` = 객체의 인스턴스. 자기자신을 참조



```

111 class Calc(object):
112     def __init__(self, num1, num2):
113         print(num1,num2)
114
115     def sum():
116         return num1 + num2
117
118 C1 = Calc(20,10)
119 print(C1.sum())

```

실행

제출

최종 제출 점수
0 점

최종 제출 시간
2023. 06. 26. 오후 01:58:54

/* 코드가 실행되는 중입니다... */

File "main.py", line 116

```

    return num1 + num2
           ^

```

TabError: inconsistent use of tabs and spaces in indentation

```

class Calc(object):
    def __init__(self, num1, num2):
        print(num1,num2)
        self.n1 = num1
        self.n2 = num2

    def sum(self):
        # return num1+num2
        return self.n1 + self.n2

```

```

C1 = Calc(20,10)
print(C1.sum())

```

- 모듈 : 함수나 변수 또는 클래스를 모아 놓은 파이썬 파일

예) aaaa.py 파일에서 abc Class가 있다면, 나는 main.py에서 작업중 .

파일 내부에 라이브러리 작성시 from aaaa import abc 이용하여 함수들 사용가능

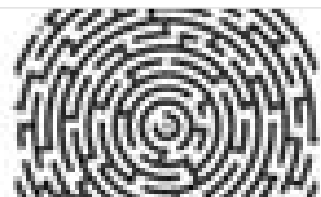
예제) 청사과, 아오리사과, 부사사과, 홍옥사과 (인스턴스:객체) ⇒ 클래스 : 사과

- 연습문제 6-3 에서 13번 문제 풀어봅시다.

연습문제 6-1

.answer {margin-top: 10px;margin-bottom: 50px;padding-top: 10px;border-top: 3px solid LightGray;bo...

<https://wikidocs.net/41106>



013

다음과 코드가 동작하도록 사람 클래스를 수정하라.

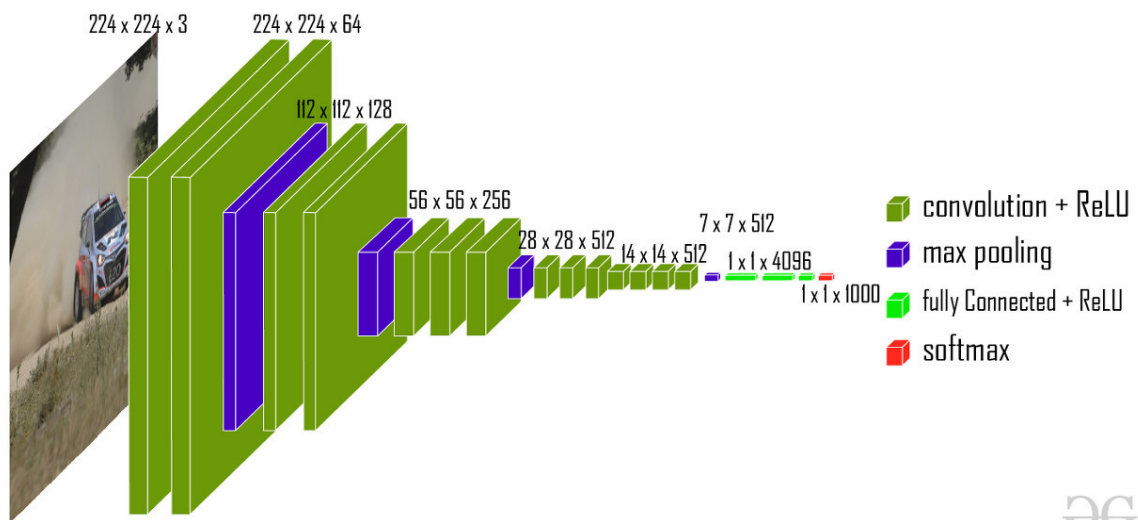
```
>> human = 사람("유종훈", 1986)
>> human.연도
1986
```

- `super()` : 상속. 부모 클래스를 불러온다. 부모-자식

CNN 모델들 (역사적으로만)

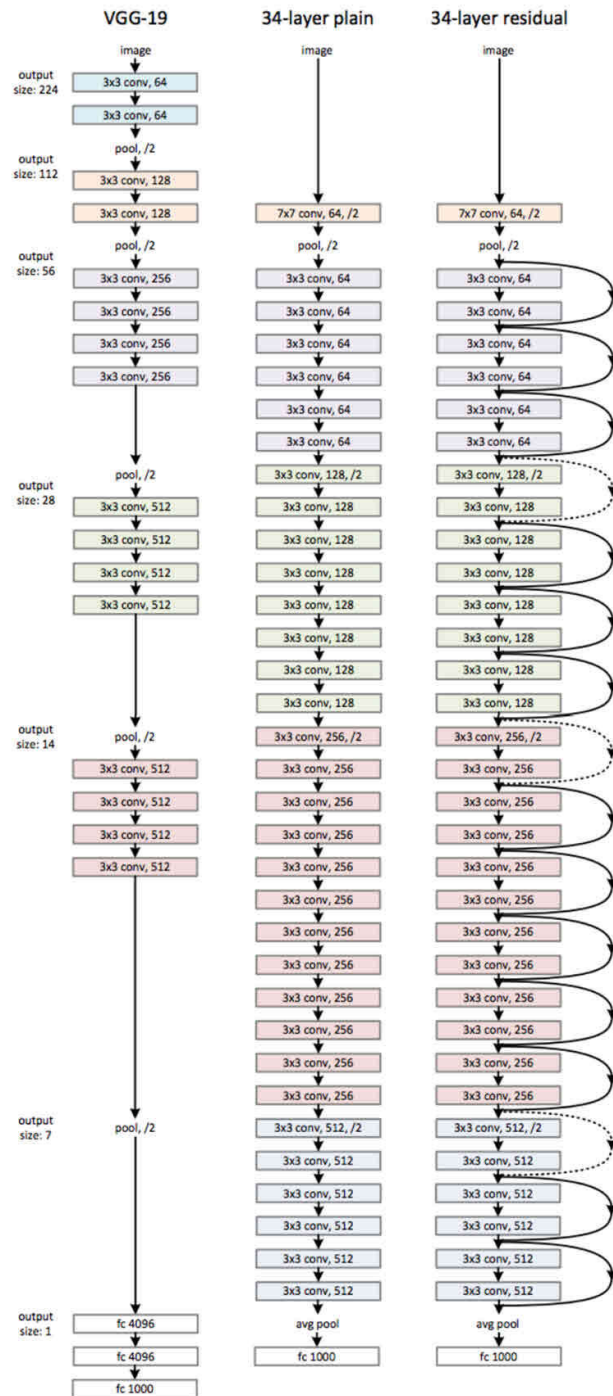
vgg 16

- 3x3 filter 사용
- 16개의 레이어를 가졌다. (vgg19는 19 layers)



Resnet

- vgg19에서 확장된 모델이며, 152 레이어까지 쌓을 수있음
- skip connection : 기울기가 1이상 \Rightarrow gradient vanishing 문제 해결!




기타 CNN 모델들

: AlexNet, GoogleNet, MobileNet+ (vgg16, resnet)

추가자료

2) VGG, GoogLeNet(레이어 22개 이하)

우리는 앞서 8 layer 이하의 비교적 적은 수의 layer를 가진 CNN model 들을 보았습니다. 지금부터 살펴볼 VGG, GoogleNet의 부터는 layer가 더...

 <https://wikidocs.net/137251>




기본 이미지 데이터들

- 프로젝트로는 coco dataset이 무난하다

MNIST, CIFAR, ImageNet, COCO

영상 인식 분야에서 많이 사용되는 대표적인 데이터셋에 대해 몇가지 소개합니다. 1.

MNIST(Modified National Institute of Standards and Technology database) 손으로 쓴 숫자들로 이루어진 대형 데이터베이스이며, 28x28 크기의 60,000개의 트레이닝 이미지와 10,000개의 테스트

 <https://page-box.tistory.com/5>

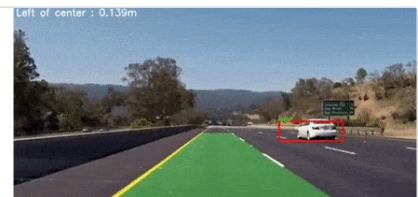


- yolo 를 이용해서 영상딥러닝 프로젝트 하시는것도 추천 (전이학습(transfer learning)

Object Detection And Lane Finding GIF | Gfycat


Watch and share Object Detection And Lane Finding GIFs on Gfycat

 <https://gfycat.com/spotlessmaturebactrian>



4K YOLO 9000 Object Detection #7 GIF | Gfycat


Watch and share Tensorflow GIFs and All Tags GIFs on Gfycat

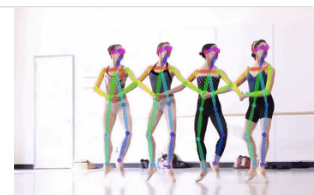
 <https://gfycat.com/immaterialbareeasternglasslizard>



Realtime Multi-Person 2D Human Pose Estimation Using Part Affinity Fields, CVPR ... | Gfycat

Watch and share Realtime Multi-Person 2D Human Pose Estimation Using Part Affinity Fields, CVPR 2017 Oral GIFs on Gfycat

 <https://gfycat.com/diligentidlequineapig>




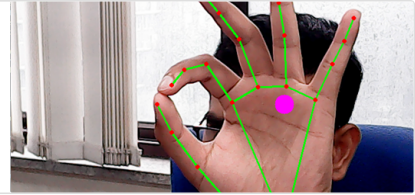
- openCV : 빠르고 간단하게 눈코입인식 (마스크 착용 유무)

<https://www.youtube.com/watch?v=yPTLc6FmQCM>

Hand joint detection using OpenCV and MediaPipe | SHAHINUR

MediaPipe is a cross-platform framework for building multimodal applied machine learning pipelines. MediaPipe Python package is available on PyPI for Linux, macOS, and Windows. Today we will...

 <https://shahinur.com/en/hand-joint-detection-using-opencv-and-mediapipe/>



퀴즈 (10분)

1. CNN에서 정사각형으로 사용되며, feature extract에 사용되는 것을 무엇이라고 하는가?
2. $\begin{bmatrix} 2 & 3 & 2 \\ 1 & 0 & 1 \end{bmatrix}$ 와 $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ 으로 dot product 연산하게된다면 결과는?
 $\begin{bmatrix} 1 & 0 & 2 \\ 1 & 0 & 0 \end{bmatrix}$ $\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$
- 2-1. 2번에서의 convolved feature의 크기는 ?
3. convolved feature의 크기가 줄어드는것 방지 + 가장자리의 정보손실 방지를 위해서 어떻게 할 수 있는가?
4. CNN모델을 아는데로 3개 나열하시오
5. VGG19는 어떤 레이어들의 갯수가 19개인가? (2 가지 종류의 레이어)
6. 학습한 모델의 요약정보를 보고싶을때 사용하는 함수는?
7. 객체들이 말티즈, 요크셔테리어, 시바견, 비송, 시츄 일때, 클래스로 적절한것은?
8. hidden layer들의 activation 함수는? 이유는?
9. 클래스에서 __init__의 역할은?
10. test.py에서의 클래스 Mymodels에는 def vgg16, def vgg19, def resnet, def alexnet이 존재한다.
main.py 파일에서 alexnet 함수를 사용하려면 어떻게 코드를 작성해야하는가 ?