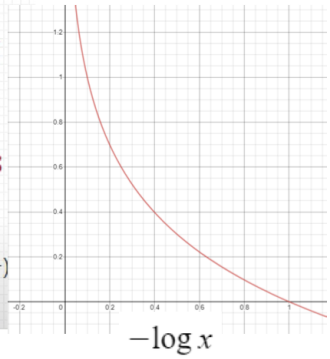
ML workspace_04

중요 개념 복습 Cross entropy loss 함수를 사용하는 이유

- **Entropy** : 정보를 최적으로 인코딩하기 위해 필요한 bit의 수

  ex) 오늘이 무슨 요일인지 bit로 전송한다. → 3bit 필요
  월: 000, 화: 001, 수: 010, 목:011, 금: 100, 토: 101, 일: 110

$\log_2 N$



- **만약 각각의 발생 확률이 다르다면?**

  ex) 40개의 문자 (A,B,C,D,...,Z,1,2,3,...14)를 bit로 전송한다. $\log_2 40 = 5.3$

  그런데 A,B,C,D가 전체의 22.5%씩 전체 90% 확률로 발생한다

  1st bit : A,B,C,D 인지 아닌지 YES → 추가로 2bit 더 필요 (ABCD 구분)
  NO → 추가로 6bit 더 필요 ($\log_2 36$)

  그러므로, 0.9 x 3 + 0.1 x 7 = 3.4 bit

- **Entropy는 각 label들의 확률분포의 함수!**

$$H(y) = \sum_i y_i \log \frac{1}{y_i} = -\sum_i y_i \log y_i$$

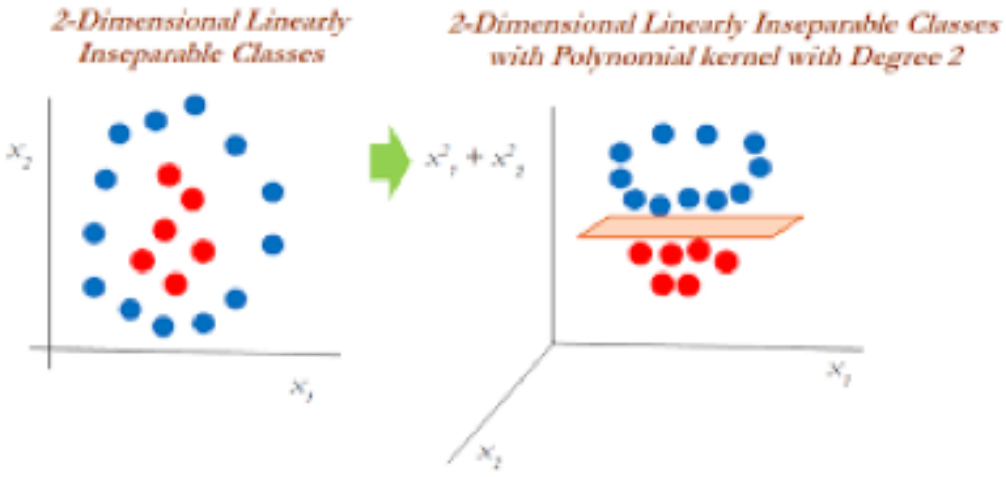
확률 information gain

$$\begin{aligned} H(TAG) &= -\,(4 * (.225 * \log_2.225) + 36 * (.0028 * \log_2.0028)) \\ &= -(-1.04 + -.82 + -.85) \\ &= 2.72 \end{aligned}$$

[참고] Terry TaeWoong Um 유튜브 채널

엔트로피는 W
확률과 반비례하다

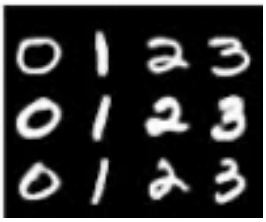
행렬곱으로 변환한 행렬을 활성화 함수룰 통해서 변환한다.
활성화 함수는 딥러닝 네트워크에서 비선형 변환을 담당한다.



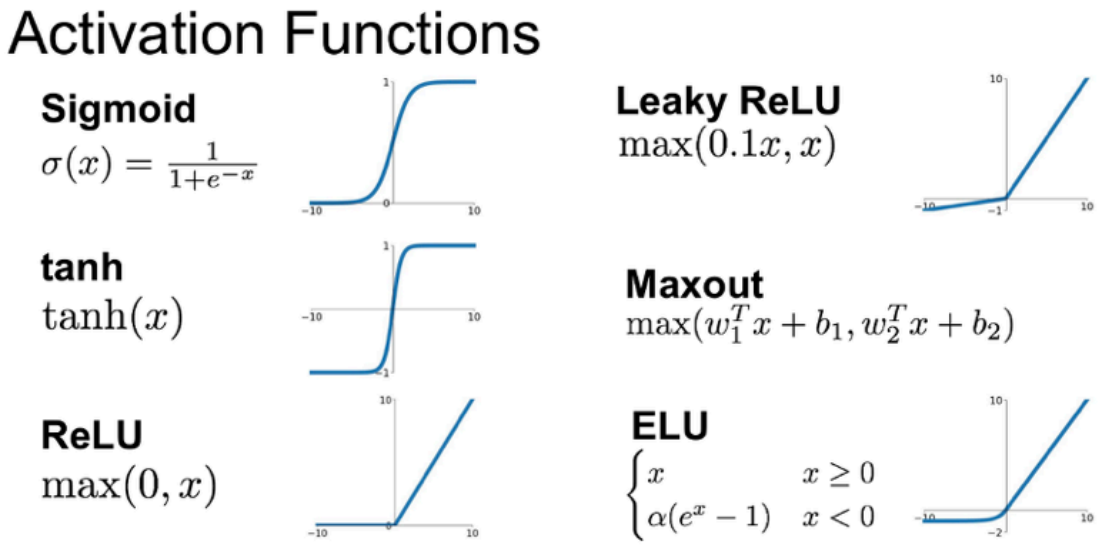
one epoch은 학습 한바퀴
banch size는
iterations는 banch size를 얼마나 반복하는지를 결정

학습시간을 효율적으로 사용하기위해 banch size를 사용

적게만든 banch size가 정규분포를 이룬다면 많은 데이터가 필요없다.

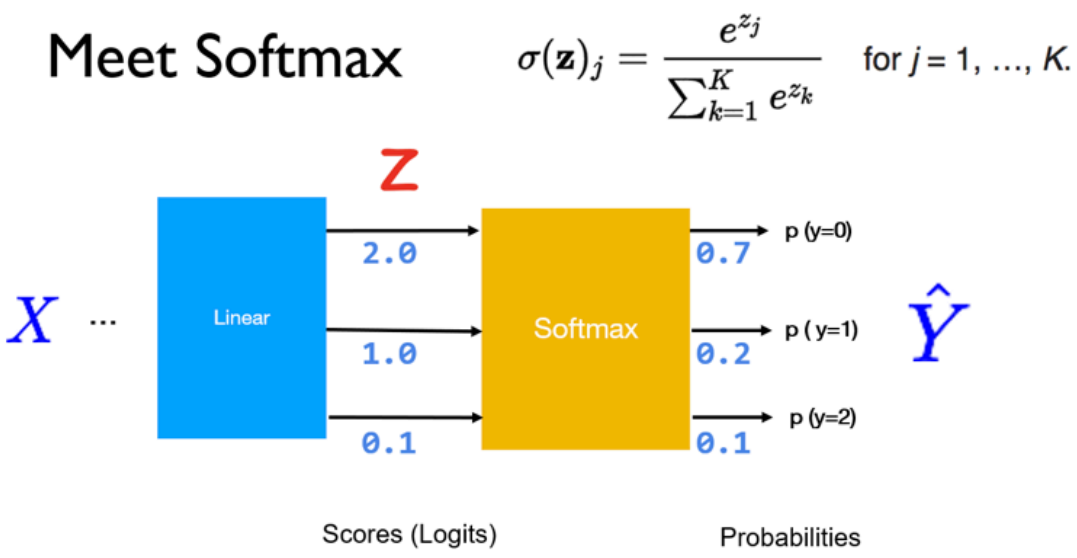


활성화 함수 종류



ReLU는 기울기전파에 유리하다

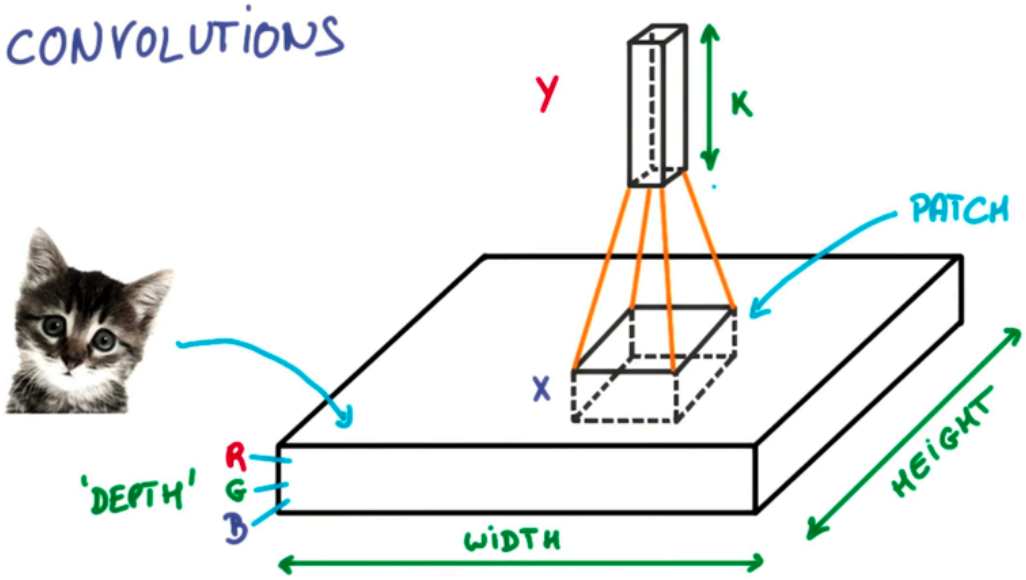
SOFTMAX 함수



SOFTMAX 함수는 Cross Entropy Loss를 구하기 위해 필요하다.
Soft max스코어를 받으면 0~1의 확률분포로 만들어준다.

1) CNN (Convolutional Neural Network)

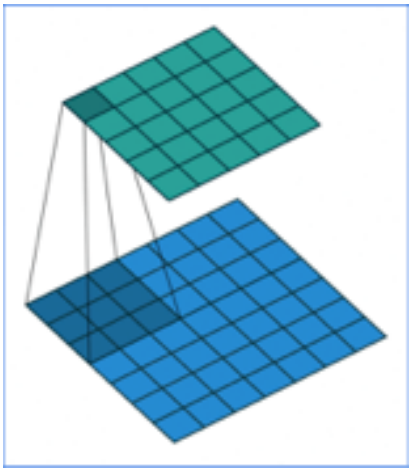
CNN: 합성곱 신경망은 이미지 처리에 탁월한 성능을 보이는 신경망이다.



2) Convolution

합성곱 연산을 통해서 **이미지의 특징을 추출**하는 역할을 한다.

커널을 이미지의 처음부터 끝까지 훑으면서 겹쳐지는 부분의 각 이미지와 커널의 원소의 값을 곱해서 모두 더한 값을 출력으로 한다.
커널(kernel)은 일반적으로 3 × 3 또는 5 × 5를 사용합니다.



3) 패딩(Padding)

위의 예에서 5 × 5 이미지에 3 × 3의 커널로 합성곱 연산을 하였을 때, 스트라이드가 1일 경우에는 3 × 3의 특성 맵을 얻었습니다. 이와 같이 합성곱 연산의 결과로 얻은 특성 맵은 입력보다 크기가 작아진다는 특징이 있습니다. 만약, 합성곱 층을 여러개 쌓았다면 최종적으로 얻은 특성 맵은 초기 입력보다 매우 작아진 상태가 되버립니다. 합성곱 연산 이후에도 특성 맵의 크기가 입력의 크기와 동일하게 유지되도록 하고 싶다면 패딩(padding)을 사용하면 됩니다.

4) 풀링(Pooling)

일반적으로 합성곱 층(합성곱 연산 + 활성화 함수) 다음에는 풀링 층을 추가하는 것이 일반적입니다.
풀링 층에서는 특성 맵을 다운샘플링하여 특성 맵의 크기를 줄이는 풀링 연산이 이루어집니다.

풀링을 사용하면, 특성 맵의 크기가 줄어드므로 특성 맵의 가중치의 개수를 줄여줍니다

풀링 연산에는 일반적으로 **최대 풀링(max pooling)**과 **평균 풀링(average pooling)**을 사용한다.

풀링 연산에서도 합성곱 연산과 마찬가지로 커널과 스트라이드의 개념을 가집니다. 위의 그림은 스트라이드가 2일 때, 2 x 2 크기 커널로 맥스 풀링 연산을 했을 때 특성맵이 절반의 크기로 다운샘플링되는 것을 보여줍니다. 맥스 풀링은 커널과 겹치는 영역 안에서 최대값을 추출하는 방식으로 다운샘플링합니다.

다른 풀링 기법인 평균 풀링은 최대값을 추출하는 것이 아니라 평균값을 추출하는 연산이 됩니다.
풀링 연산은 커널과 스트라이드 개념이 존재한다는 점에서 합성곱 연산과 유사하지만, 합성곱 연산과의 차이점은 학습해야 할 가중치가 없으며 연산 후에 채널 수가 변하지 않는다는 점입니다.