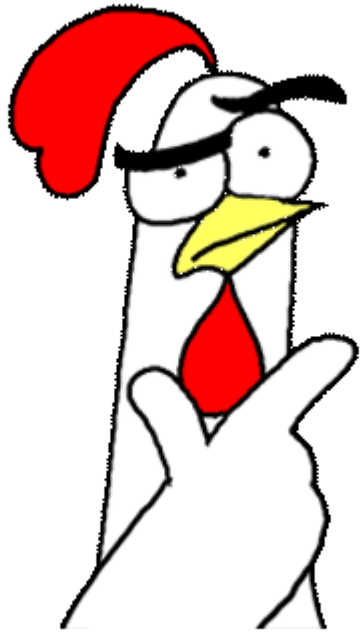


밑바닥부터 시작하는 딥러닝

Chapter 4.2 손실 함수

박정현

손실 함수 : 다름의 정도를 계산하는 함수



평균 제곱 오차

$$E = \frac{1}{2} \sum_k (y_k - t_k)^2$$

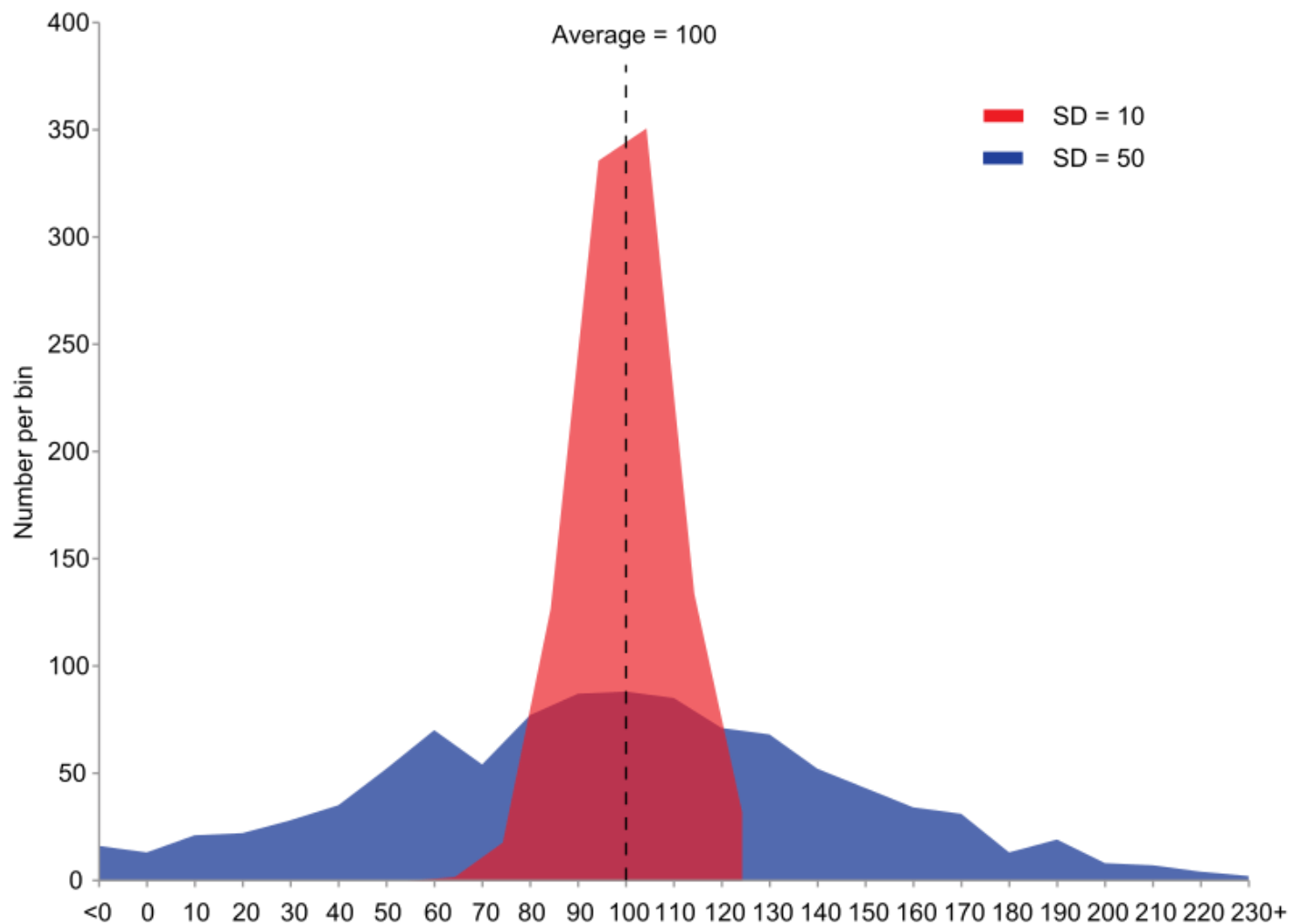
분산을 의미한다.



분산이란?

- 데이터가 평균에서 떨어져 있는 정도를 나타낸다.

-wikipedia



평균 제곱 오차

나눠주는 수는 상관 없음!

값이 너무 커질 것 같으면 사용자 편의상 사용 가능!

$$E = \frac{1}{2} \sum_k (y_k - t_k)^2$$
$$\frac{1}{n} \sum_{i=1}^n (y_i - t_i)^2$$

평균 엔트로피 오차

$$-\sum_{i=1}^n \ln(y_i) * t_i$$

정보 엔트로피와 연관!



확률이 높을수록 정보량은 낮다.

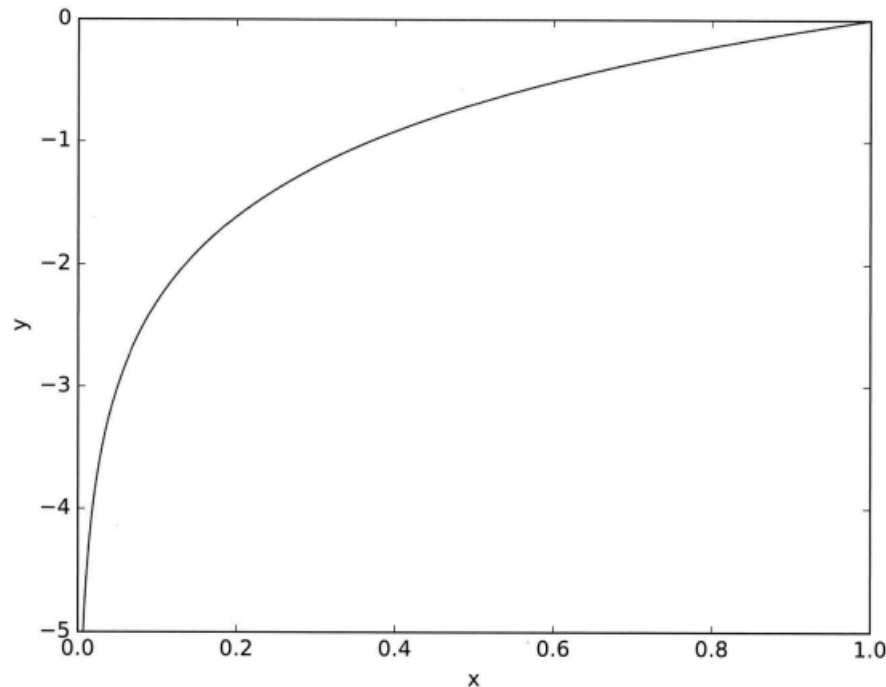
평균 엔트로피 오차

$$-\sum_{i=1}^n \ln(y_i) * t_i$$

y는 0부터 1까지의 값을 가지는 확률로 인식할 수 있다.

이 때 $\ln(y)$ 의 그래프를 보면 y가 1에 가까워질수록 $\ln(y)$ 의 값은 0에 가까워진다는 것을 알 수 있다.

그림 4-3 자연로그 $y = \log x$ 의 그래프



평균 엔트로피 ↓ == 정답일 확률 ↑

배치처리 시 엔트로피 오차 구현

```
def cross_entropy_error(y, t):  
    if y.ndim == 1:  
        t = t.reshape(1, t.size)  
        y = y.reshape(1, y.size)
```

왜 y.ndim == 1일 때 reshape를 해주나?

```
    batch_size = y.shape[0]  
    return -np.sum(t * np.log(y)) / batch_size
```

배치처리 시 엔트로피 오차 구현

```
def cross_entropy_error(y, t):  
    if y.ndim == 1:  
        t = t.reshape(1, t.size)  
        y = y.reshape(1, y.size)
```

```
    batch_size = y.shape[0]  
    return -np.sum(t * np.log(y)) / batch_size
```

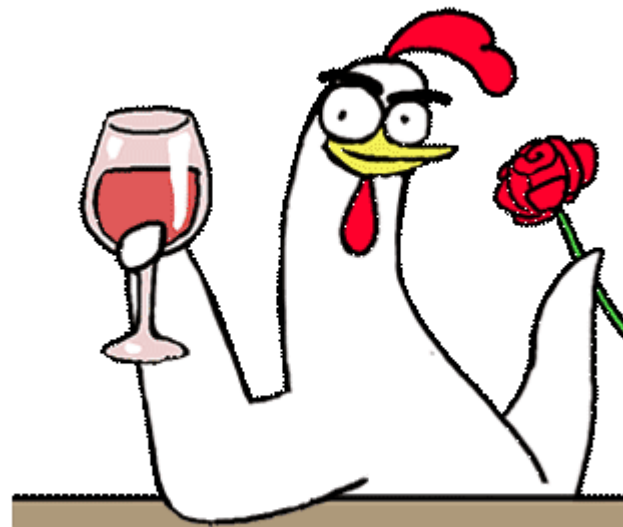
왜 y.ndim == 1일 때 reshape를 해주나?
일반화하기 위해서

```
y = np.array([0, 1, 2, 3])  
print(y.shape)  
>> (4,)
```

학습의 지표

정확도 = (맞은 개수) / (전체 개수)

손실함수 = 평균 엔트로피 혹은 평균 제곱



학습의 지표

정확도 = (맞은 개수) / (전체 개수)

매개변수를 조정해서 정확도를 변경
시키기 쉽지 않음.

손실함수 = 평균 엔트로피 혹은 평균 제곱

미세한 매개변수의 조정도 큰 변화를
가져올 수 있음.

학습의 지표

정확도 = (맞은 개수) / (전체 개수)

손실함수 = 평균 엔트로피 혹은 평균 제곱

매개변수를 조정해서 정확도를 변경
시키기 쉽지 않음.

미세한 매개변수의 조정도 큰 변화를
가져올 수 있음.

Ex) 매개변수에 + 혹은 - 0.1을 준다.

정확도 변할수도 안변할수도 있음.

손실함수의 값이 증가하거나 감소

