

## Cs231n Lecture 3 Summary

### Loss Functions and Optimization

#### 1. Loss function

Lecture 2 에서  $w$  값을 이용한다고 했을 때, score 에 영향을 주는  $W$  값을 정량화할 수 있는 것이 필요하다. 이것이 loss function 이다. Loss 를 구하는 방법은 각 카테고리 별로 loss 를 구해 더한 뒤 카테고리의 개수로 나눠 평균값으로 만들어 준다.

##### 1) SVM loss

SVM loss 는 서포트 벡터 머신의 줄임말로, 힙지 로스(hinge loss)라고도 불린다. 만약 정답인 클래스가 정답이 아닌 클래스보다 safe margin 이 크면 loss 를 0 이라고 간주한다. 이 loss 의 특징은 데이터에 둔감하다는 것이다. 즉 구체적인 score 의 점수에는 관심이 없고, 단지 정답 클래스가 다른 클래스보다 높은지에만 관심이 있다. 그리고 loss 의 최솟값은 0, 최댓값은 무한대이다.

만약  $w$  가 매우 작아져서 score 이 0 에 가까워진다면, loss 값은 항상 (클래스 개수-1)의 값이 나온다. 이 방법은 디버그 용으로 많이 사용하는데, sanity check 라고도 부른다.

지금까지 계산할 때 정답 클래스의 값은 빼고 계산하였는데, 만약 정답 클래스까지 포함시켜 계산한다면 loss 값의 평균이 +1 이 되므로 0 에 가까워지지 않는다. 이것이 정답 클래스를 제외시키는 이유이다.

만약 전체 값을 제공해서 구한다면 super hinge loss 방식이 되는데, non linear 하기 때문에 실제로는 잘 사용하지 않는다.

하지만 이 경우에는  $w$  값이 여러 개가 될 수 있기 때문에 학습 과정에서 overfitting 이 발생할 수 있다. 이때 필요한 것이 규제(regularization)이다. 이 규제 값을 감안하여 최적의  $w$  값을 찾아낸다. L1 regularization, L2 regularization 방식이 있다.

## 2) Softmax Classifier

다른 말로는 크로스 엔트로피라고도 한다. 모든 스코어값을  $\exp$  취하고 더한 뒤 원하는 클래스의 점수를  $\exp$  취해서 나눈다. 그러면 확률값이 나오게 된다. 그리고 내가 원하는 정답 클래스에  $-\log$  를 취해서 loss 를 구한다.

최솟값은 0, 최댓값은 이론적으로 무한대이다.

만약 스코어가 0 에 가까워진다면, Loss 값은  $-\log(1/\text{클래스 개수})$  가 된다.

이것도 마찬가지로 디버그용으로 사용된다.

크로스 엔트로피 방식은 확률로 데이터를 계산하기 때문에 조금만 변경되어도 큰 차이를 나타낸다. 즉 데이터에 민감하다.

## 2. Optimization

좋은  $w$  값을 찾는 과정이다. Loss 가 0 인 지점을 찾는 것과 같다. 많이 사용하는 방법은 gradient descent, 즉 경사 하강법이다. 하나하나 미분을 해서 수치적으로 구할 수는 있지만, 값이 많아진다면 시간이 오래 걸리게 되어 효율이 떨어진다. 그래서 해석적 방법(analytic gradient)을 사용한다.