

Image Classification

↪ A core task in Computer Vision

고양이 사진 given to 컴퓨터 : Just a big grid of numbers b/w 0-255

Challenges ① Viewpoint variation ② Illumination
③ Deformation ④ Occlusion
⑤ Background Clutter ⑥ Intradass variation

"Data-Driven Approach"

1. Images & Labels dataset 수집
2. Use Machine learning to train a classifier
3. Evaluate the classifier on new images

1st classifier = Nearest Neighbor

train - Memorize all data and labels

predict - Predict the label of the most similar training img.

Train O(1), Predict O(N) : 훈련은 브라아빠서 좋은 방법은 아님.

* L1 distance : $d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$ 맨해튼 거리

이미지 간 비교 방법

I_1 의 픽셀 - I_2 의 픽셀 이 절대값 쓰운 각 값을 sum

K - Nearest Neighbors

가장 가까운 이웃으로부터 label 가져오는 대신

K_m closest points로 **majority vote** 통해 결정

* L2 distance : $d_2(I_1, I_2) = \sqrt{\sum_p (I_1^p - I_2^p)^2}$ 유클리드 거리

K or distance 치적 "직접" 정해야 하는 것? Hyperparameters

data와 problem에 따라 성능 다른

Setting hyperparameters

Idea 1 Your Data의 좋은 성능

BAD: 항상 $K=1$ or perfect 성능

Idea 2 Split train / test,

test set이 좋은 성능

BAD: 세트간 데이터 성능 보장 불가

Idea 3 Split train / Validation / test

Val set이 좋은 성능

Better, test set은 과거학습에 딱한번 사용

Idea 4 Cross - Validation

Split data into folds, try each fold as validation

and average the results (보통에서 잘 안쓰임)

KNN on images never used

- Very slow @ test time
- Distance metrics on pixels are not informative
- Curse of dimensionality 차원증가

Linear Classification

Parametric Approach

Image



$32 \times 32 \times 3$ bytes

파일 크기

$f(x, W) \rightarrow$ 10 numbers giving

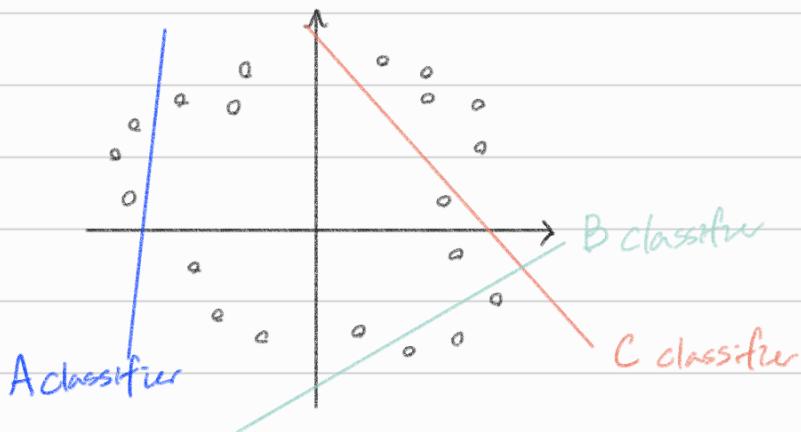
parameters of weight

class scores

$$f(x, W) = Wx + b$$

↓ 10x1 ↓ 3072x1
(32x32x3 = 3072) 10x3072 3072x1
 10개 블록에 대한 정수 ↓ f(x, W) ↓ 10x1 }
 } Stretch pixels
 into column

Example trained weights of a linear classifier



but 선형분류로 분류하기 어려운 types 존재