

Component

- Normalization Layer
- Dropout

Norm

Dropout (Regularization & CAM)

- Mối ép ko phụ thuộc vào 1 Neuron
- Tài cơ' phải độc lập
- Tránh overfitting

chỉ sử dụng để train / test thì ko dùng

có 1 unit sẽ ko train đủ 6 người

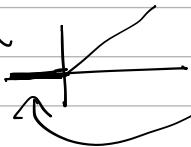
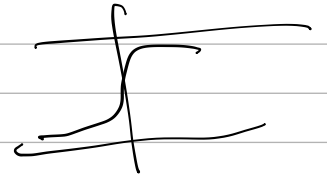
test thì 10 người

⇒ cần phải scale nó giảm xuống

Activate function:

Sigmoid tăng common
bây h là relu
hay là

gi' mà $+\infty$



nó vẫn bị flat ở đây

→ nó thường nằm ngay sau linear layer

case study: VGG Net

thì tại sao người ta sử dụng 3×3 conv?

Sao ra được con số này

thì người ta giải thích là đang dùng 1×1 conv
thì dùng như là bộ lọc nhỏ xếp chồng lên nhau thì hay hơn
vừa giúp máy như là receptive field vừa tiết kiệm bộ
nhớ, vừa có như là lớp học như là đặc trưng phức tạp.

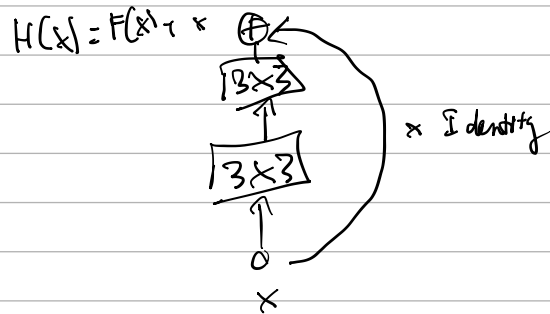
ResNet

Thì người ta đặt câu hỏi là sao ko dùng như là lớp học?
Tại sao 66 layers vs 20 layers lại khác?

thì cần tạo ra là ko phải giải học mà optimize nó khác

y' là như lớp thì luôn tốt hơn nhưng mà phải optimize được.

thì ResNet có gọi y' là sử dụng residual block.



- Như thế nào để gọi mà double filter, ...
khi có lớp conv ở đầu tiên của block.

How to train

- weight initialization case: giá trị quá nhỏ?
nhỏ vậy thì khi nhân thì kết quả sẽ tiến về 0

Solution:

Kaiming initialization:

$$\text{với } \sqrt{\frac{2}{\text{Dim}}}$$

→
Cố Node
ở đầu tiên

Data Augmentation

- Horizontal Flipping: cơ bản là lật cái hình lại thôi

- Resizing and cropping: chắc nên nói kỹ quy trình 3 bước của ResNet.

→
với Test time Augmentation: thay cái này khác hay

→
độ chính xác cao

→
bù lại thì chậm hơn do chạy nhiều lần.

- color jitter:

Cut Out: Theo mình hiểu thì là tạo ngẫu nhiên
những vùng rồi làm cho nó đen (hoặc xám)
rồi train → còn test thì use full image

Question: Nếu như ko có nhều Data, liệu có thể train CNNs

TH: ít Data) mình có thể chọn các mô hình đã pretrained rồi Fine-tune
mình sẽ lớp cuối rồi (ở đây là lớp Fully connected)
khởi tạo bằng kaiming initialization.

Data nhều) còn nếu như mình có nhều Data, thì Unfreeze tất cả?
phải có bias
dừng lại những giá trị của model, khi đó training sẽ nhanh hơn

Thích DataSet này mà ít quá) Tìm DataSet lớn tương tự, train model lên trên DataSet đó
Rồi Transfer learn lên DataSet của mình.

Hyperparameter Selection

Step 1: KT loss khởi tạo có hợp lý ko

Step 2: Cố định làm cho nó overfit

Step 3: Tìm LR sao cho loss giảm

Step 4: chọn hyperparameter cách lên VD: 1×10^{-1} , 1×10^{-2} , 1×10^{-3} ...
nếu là do thấy bằng cách RS RST ~ 1.5 epoch.

Step 5: thu hẹp phạm vi lại train longer

Step 6: KT loss, Accuracy