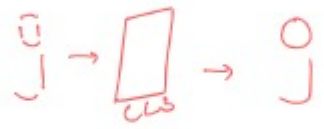
CNN

* Architecture:-
① Convo → Feature Extraction

Edge → change in intensity
Feature → Jisse apki image bani hoti

MNIST
↓
0 - 9



* Filters/Kernel
↳ Feature Ext

* Edge Detection

6×6 matrix

n×n

3×3 Filters

m×m

=

Feature Map

(n-m+1)(n-m+1)
(6-3+1)
(4×4)

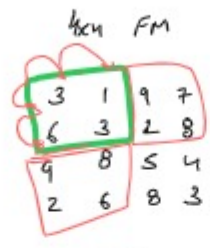
* Drawbacks
① Memory issue → 10 - 15 mB
② Translation Variance → Important
* Features → Location Dependent
M

→ Predict *

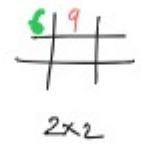
* Pooling → * Translation Invariance
Feature map → downsample → most imp Features Extract
Working

* Types of pooling
① Max Pooling → Most common used
② Min
③ Avg
④ L2
⑤ Global → G Max, G Avg

* When you apply PL you have to tell about these 3 parameters
① Size → (2,2)
② Stride → 2
③ Type → max



① Size → (2,2)
③ type - max 2×2



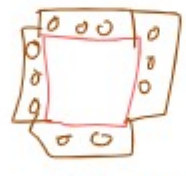
② Stride → 2
↳ Skip step
Min = 2

Pooling: n×n m×m
Stride = $\left(\frac{n-m}{s}+1\right)$

* Downsampling → ✓ ✗ ?
→ FM → Info →

* Padding

* Implementation
① Same →
② Valid
↓
XP



6×6 → 8×8

P & S

→ (n+2P-M+1)
(6+2(1)-3+1)
= 6×6

$\left[\frac{n+2P-m}{s}+1\right]$

CNN → CL → Loss → ① BCL → Log Loss
② MCL → Softmax

Relu → Adam, Adamax