# Q/A Session

Performance :    Amount of compute resource          Tabular : XGBoost

                                                                                    CPU

CY Models :    1 GPU      RTX        RTX        Apple M3      CPU+NPU
                          4080       3070

                           A6000      A5000  → training

Language Models (Transformer)  { BERT →  ↗  A6000, A5000

                                 GPT ; GPT2 → A6000
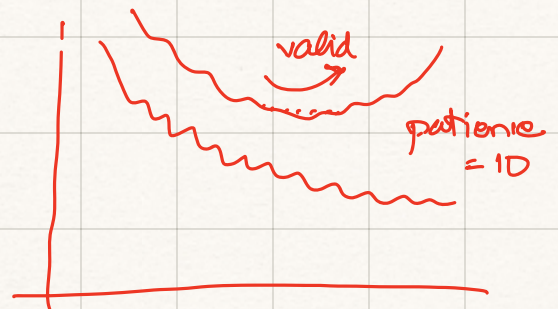
                                 GPT3+    → Training    } A100
                                            Fire tuning

                                                        multiple
                                                        A100
Diffusion: Gen Image :    multiple A100 ; H100      Training

At inference : MLDPS →  CPU is good :  GPU RTX 4080

    API for  LLM → Browser

    Memory → Batch size

    pipeline



valid

patience
= 10

model. fit()                     model

         ↖                       Layers → weights
    call back

                  beg iter       Layers (batch norm) → mean
                  end epoch                             std dev.

batch norm,                                              Dropout
           ∘ ]                                           Regularized
           ∘ ]
Dense (10) ∘ ]                                           training
           ∘ }
 .6        ∘ }                                        → Augmentate
           ∘ } ⊗
           ∘
           ∘
           ∘

Transfer Learning          Conv Base, trainable = False

Fit

iter 1   2   3   4   5   6  7  8  9  10   . . .

$\underbrace{\qquad\qquad\qquad}_{\text{epoch 1}}$   $\underbrace{\qquad\qquad}_{\text{epoch 2}}$
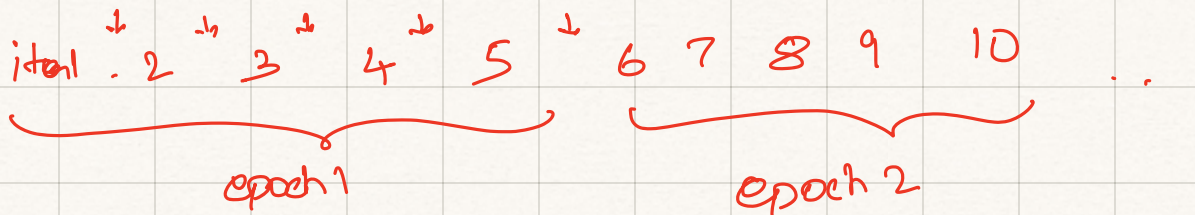
XG Boost → Tree based boosting



Sigmoid
saturating act

$\frac{\partial L}{\partial \theta} \to 0$

Deep model

← saturat
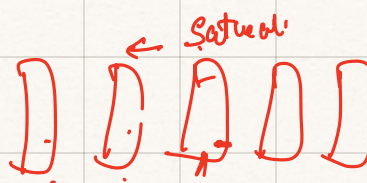


gradient → 0

back pro

batch normalizatn

XG Boost   treat cat data

sklearn pipeline
std scaler
column transformation



2x2

4x4
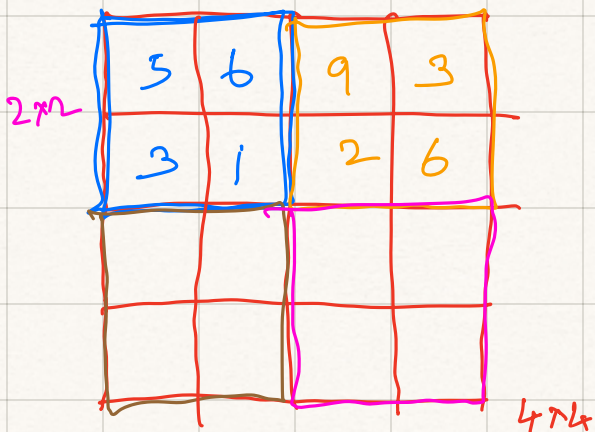
Pooling
pool = 2 → kern
$\underline{2 \times 2}$
stride = 2

6  9

pooling kernel        integral kernel
convolut kernel

LOSS

=)



early stopping

epoch

learning rate

Global minima

$\theta_2$

Full batch
gradi.

$\theta_1$
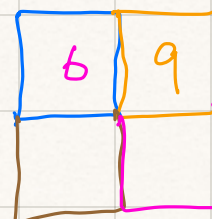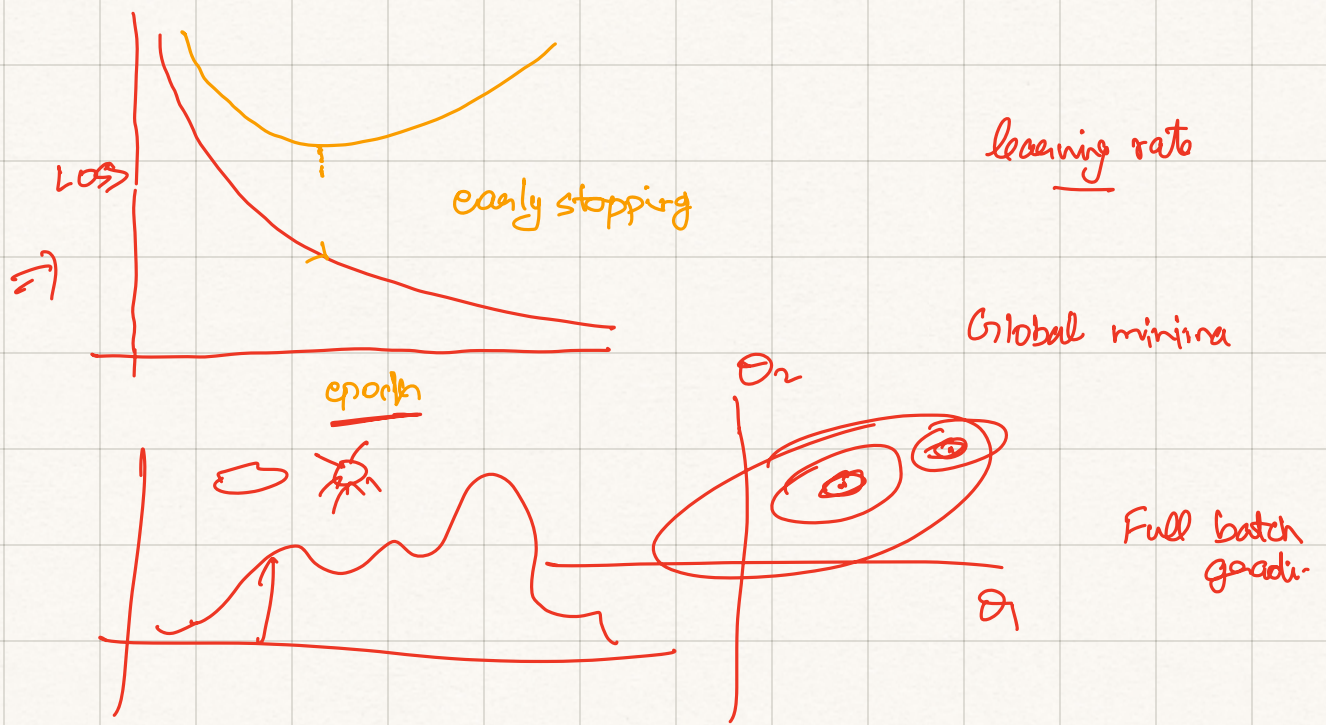
① Dropout -

② Regularization ⟹ norm $L1, L2$

5 min

Fault detection :

Telemetry

✓ 1. Label data ←

✓ 2. Unsupervised

Anomaly detection

nominal
PD

predictive model

3.

|——————15 min——————|—2min—|

ARIMA , XGBoost,
RNN , Transform , FNO

time → FFT feature

Why is activation function req.

PL ○
          ○ → ○ →
PW ○

$z_1 = \theta_0 + \theta_1 PL + \theta_2 PW$

$z_2 = \beta_0 + \beta_1 z_1$

$z_2 = \beta_0 + \beta_1 [\theta_0 + \theta_1 PL + \theta_2 PW]$

$= \beta_0 + \beta_1 \theta_0 + \beta_1 \theta_1 PL + \beta_1 \theta_2 PW$

Act                 Loss

Sigmoid         Binary CE                    ADAM

Softmax          Cat CE

∅ -
∅

224 ↑
224        3        Large num
$10^7$      ---       of FM
                        10

                   4  ⅙
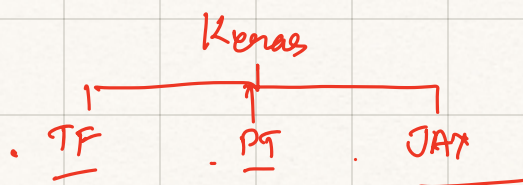                   32  64  128

                                          Classification

                                             1 ,  5

$$L1 : \quad |\theta_1| + |\theta_2| + |\theta_3| + \ldots$$

$$L2 : \quad (\theta_1)^2 + (\theta_2)^2 + (\theta_3)^2 + \ldots$$

Keras

• TF        • PT        • JAX

nn
Comp Graphs
   Auto Diff
Layers
  Training

Dev

Test