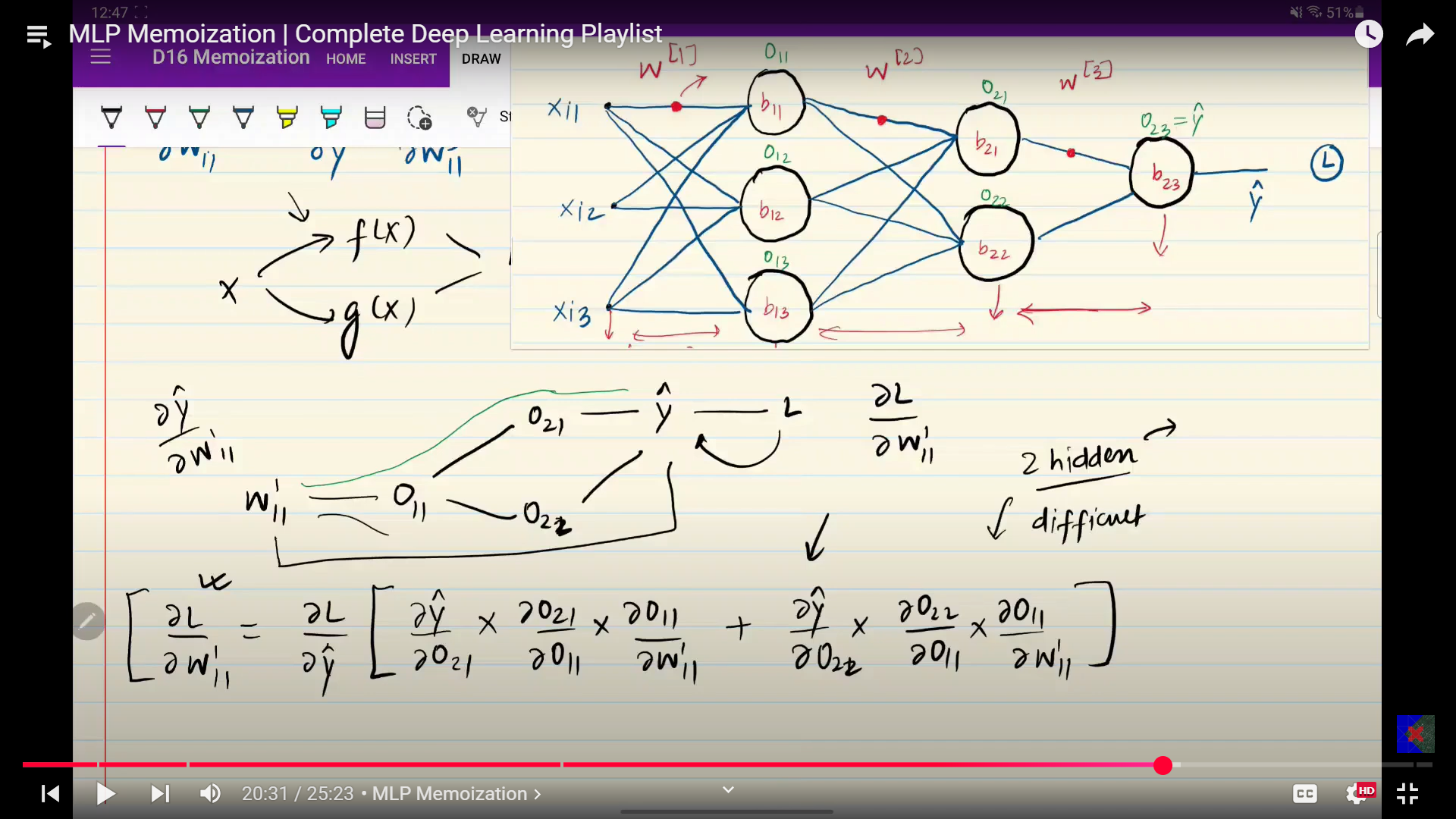
## whorizontal line

Gradient Descent (class 11)

15.01.2025

# Memoization

Memoization is an optimization technique used to speed up computer programs by storing results of expensive function calls and returning cached results when the same input occurs again.



# Types of Gradient Descents

They differ on the basis of speed , convergence .

## Batch GD

1. Take the entire dataset in 1 epoch.

2. By forward propagation take outputs (by dot product) , calculate loss

3. Update the parameters in one go.

4. Batch size = 1

No. of Updates = No. of epochs

## Stochastic GD

1. In one epoch a inner loop calls for row by row

2. Same by fp take o/p , loss

3. Update parameters one by one row . when all rows are done ,next epoch.

4. Batch size = no. of rows

No. of Updates = N \* No. of epochs

## Mini batch GD

1. If data has 320 rows, in one batch it takes 32 rows .

2. Batch size = 32

No. of Updates = No. of batches \* No. of epochs

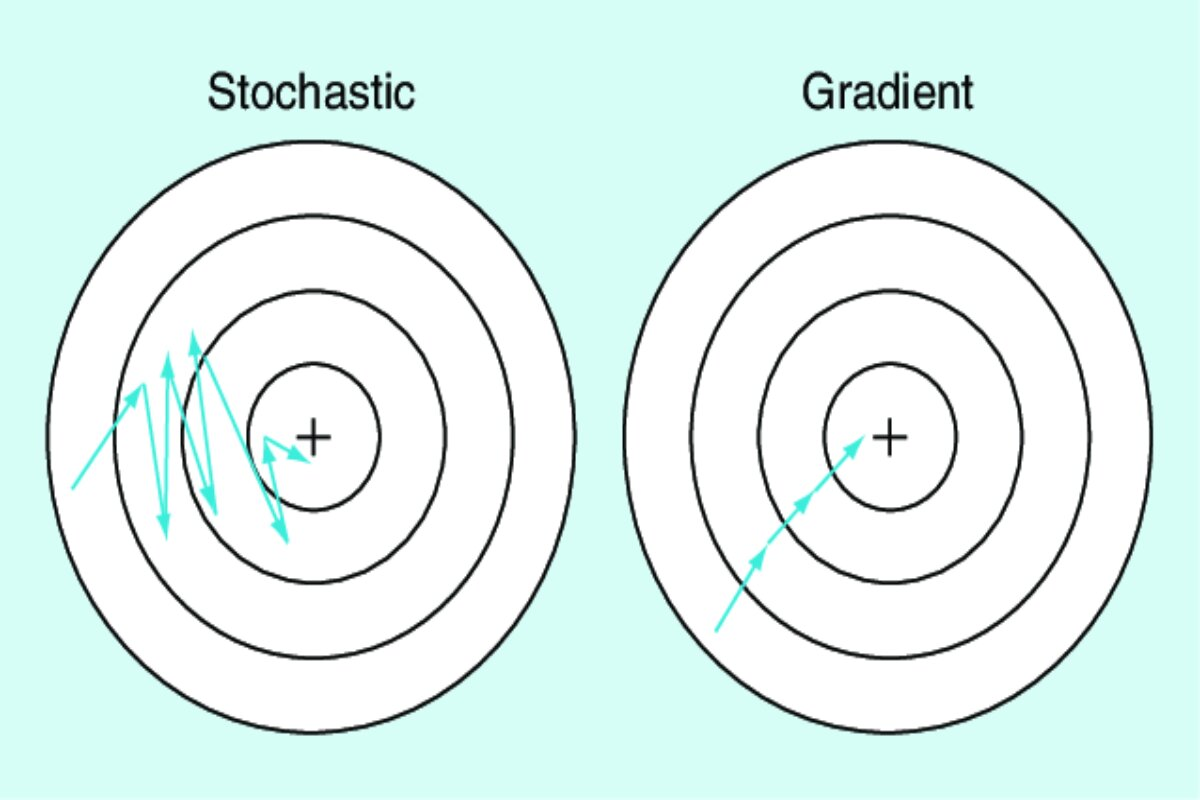
## Q1 Which is faster to give results ?

Batch (b > mini > s)

## Q2 Which is faster to converge ?

Stochastic , because no. of updates are more and actually converge before epochs completed (s > mini > b)

## Q3 Spiky Behaviour ?



Stochastic makes updates on the basis of one one row while the batch makes them on the basis of the entire dataset , that’s why it smoothly converges while stochastic shows fluctuations .

Pros - reach global minima

Cons - not exact solution

## Q4 Vectorization ?

Batch shows vectorization as a smart replacement by dot product .

## Q5 Why is the batch size in multiples of 2 ?

Here batch sizes are in exponents of 2 because that's how the GPU allocates the chunk. If you put the batch size as 100 then it would still reserve space for 128 and would only use 100.

Also, the allocated space that it's not going to use will cause issues as GPU can only reserve continuous chunks unlike CPU where it can allocate chunks which are not necessarily continuous.