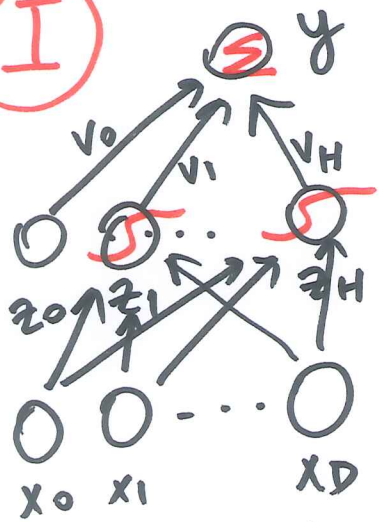
MULTILAYER PERCEPTRONS

I



$$\hat{y}_i = \underline{v}^T \cdot z_i$$

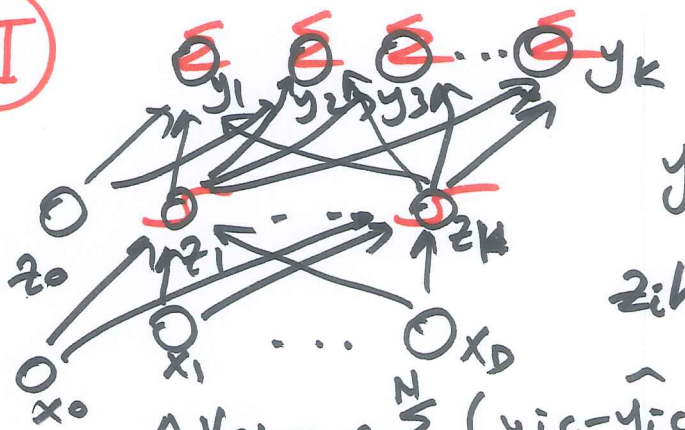
z_{ih}

$$z_{ih} = \text{sigmoid}(w_h^T \cdot x_i)$$

$$\Delta v_h = \eta \sum_{i=1}^{N} (y_i - \hat{y}_i) \cdot z_{ih}$$

$$\Delta w_{hd} = \eta \sum_{i=1}^{N} (y_i - \hat{y}_i) \cdot v_h \cdot z_{ih}(1 - z_{ih}) \cdot x_{id}$$

II



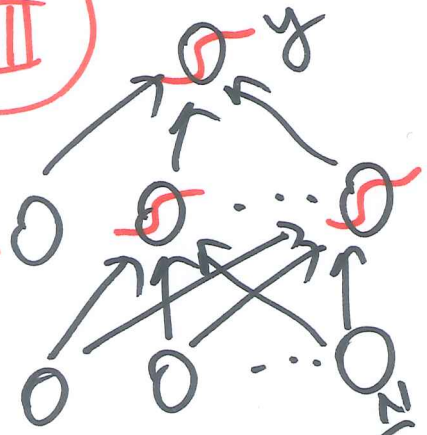
$$\hat{y}_{ic} = v_c^T \cdot z_i$$

$$z_{ih} = \text{sigmoid}(w_h^T x_i)$$

$$\Delta v_{ch} = \eta \sum_{i=1}^{N} (y_{ic} - \hat{y}_{ic}) z_{ih}$$

$$\Delta w_{hd} = \eta \sum_{i=1}^{N} \left[\sum_{c=1}^{K} (y_{ic} - \hat{y}_{ic}) v_{ch} \right] \cdot z_{ih} \cdot (1 - z_{ih}) \cdot x_{id}$$

III



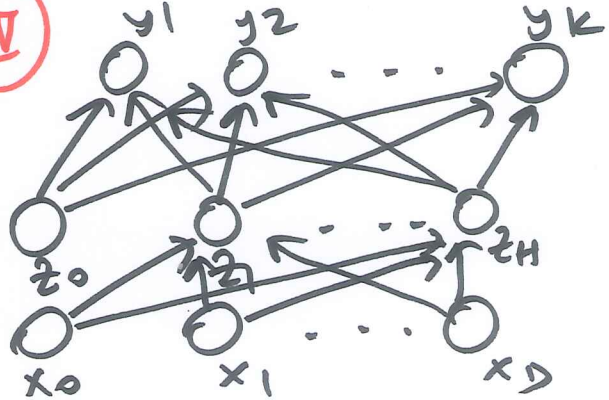
$$\hat{y}_i = \text{sigmoid}(v^T \cdot z_i)$$

$$z_{ih} = \text{sigmoid}(w_h^T \cdot x_i)$$

$$\Delta v_h = \eta \sum_{i=1}^{N} (y_i - \hat{y}_i) \cdot z_{ih}$$

$$\Delta w_{hd} = \eta \sum_{i=1}^{N} (y_i - \hat{y}_i) v_h z_{ih}(1 - z_{ih}) x_{id}$$

IV

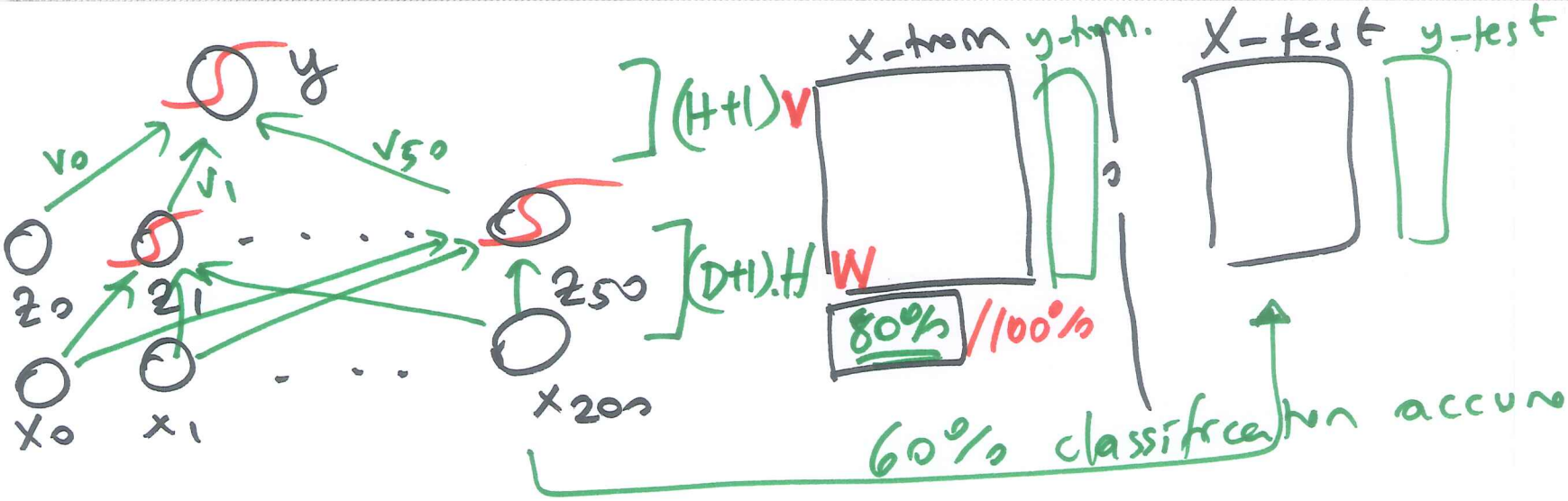


$$\hat{y}_{ic} = \text{softmax}(v_c^T \cdot z_i)$$

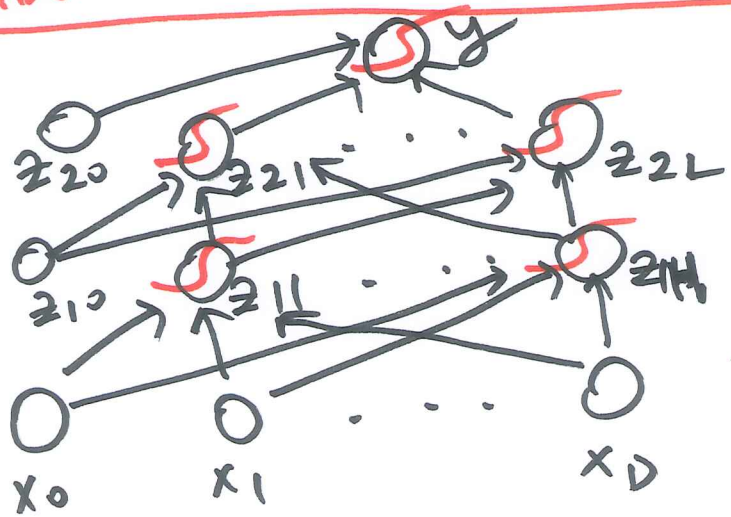
$$\hat{y}_{ic} = \frac{\exp(v_c^T \cdot z_i)}{\sum_{d=1}^{K} \exp(v_d^T \cdot z_i)}$$

$$\Delta v_{ch} = \eta \sum_{i=1}^{N} (y_{ic} - \hat{y}_{ic}) \cdot z_{ih}$$

$$\Delta w_{hd} = \eta \sum_{i=1}^{N} \left[\sum_{c=1}^{K} (y_{ic} - \hat{y}_{ic}) \cdot v_{ch} \right] \cdot z_{ih} \cdot (1 - z_{ih}) \cdot x_{id}$$



MULTIPLE HIDDEN LAYERS



$$y = \text{sigmoid}(v^T \cdot z_2)$$

$$z_{2l} = \text{sigmoid}(W_{2l}^T \cdot z_1)$$

$$z_{1h} = \text{sigmoid}(W_{1h}^T \cdot X)$$

Training Procedures

Momentum:

$$\Delta w_h^{(t)} = -\eta \frac{\partial Error^{(t)}}{\partial w_h} + \underbrace{\alpha \Delta w_h^{(t-1)}}_{\text{momentum term}}$$

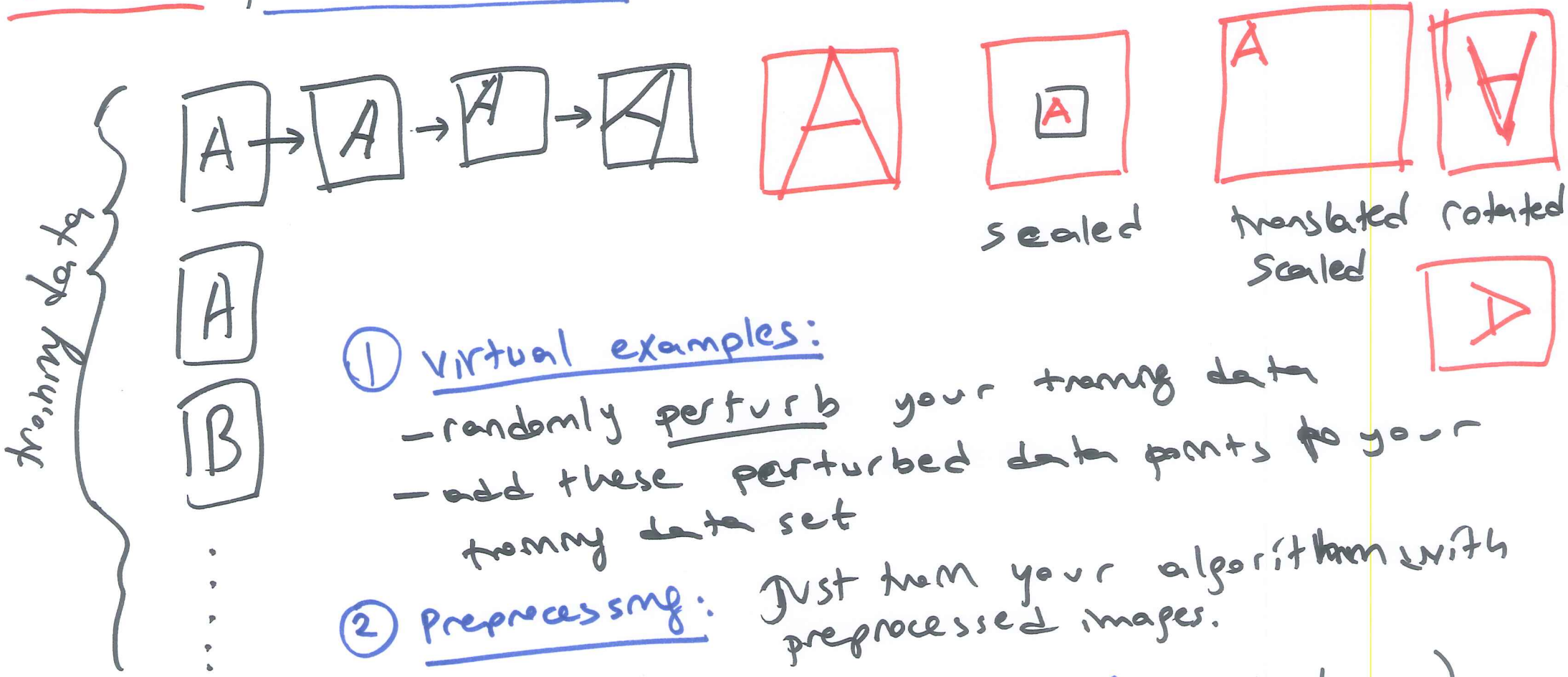
$0.5 < \alpha < 1$

Adaptive Learning Rate: $\eta \Rightarrow$ should be getting smaller in the last iterations.

Early stopping:

HINTS

prior knowledge (hints)



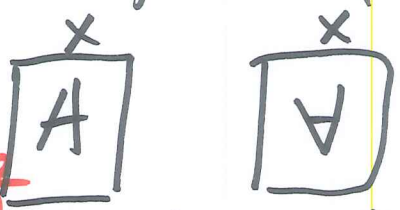
① virtual examples:
- randomly perturb your training data
- add these perturbed data points to your training data set

② Preprocessing: Just train your algorithm with preprocessed images.

③ special network structure: (weight sharing).

④ modifying error function:

$$Error' = Error + [g(x|\Theta) - g(x'|\Theta)]^2$$



weight sharing:

spatial dependency
↳ space, location

temporal dependency
↳ time.