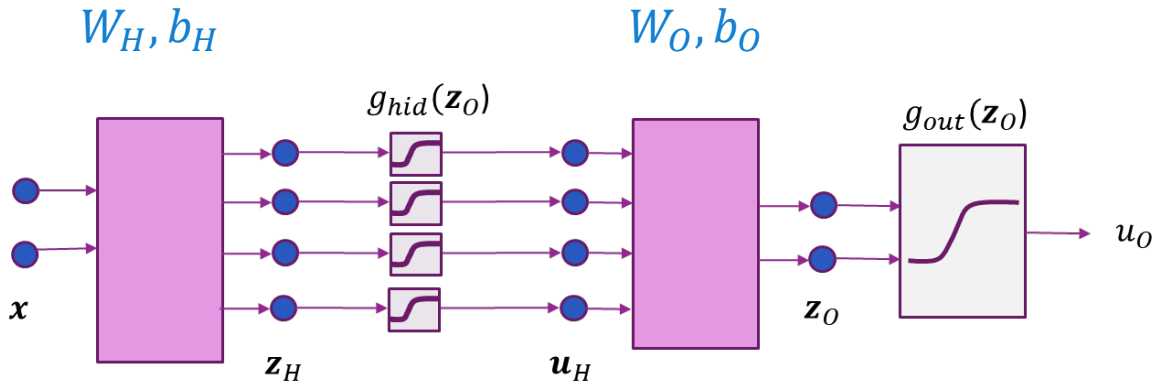
Training a Neural Network

- Given data: $(x_i, y_i), i = 1, \ldots, N$

- Learn parameters: $\theta = (W_H, b_H, W_O, b_O)$
  - Weights and biases for hidden and output layers



- Will minimize a loss function: $L(\theta)$

$$\hat{\theta} = \arg\min_{\theta} L(\theta)$$

  - $L(\theta)$ = measures how well parameters θ fit training data (x_i, y_i)

Number of Parameters

Layer	Parameter	Symbol	Number parameters	Example $N_I = 5, N_H = 20, N_O = 3$
Hidden layer	Bias	b_H	N_H	20
	Weights	W_H	$N_H N_I$	20(5)=100
Output layer	Bias	b_O	N_O	3
	Weights	W_O	$N_O N_H$	3(20)=60
Total			$N_H(N_I + 1) + N_O(N_H + 1)$	183

- Sizes:
  - $N_I =$ input dimension, N_H= number of hidden units, N_O=output dimension
- N_H= number of hidden units is a free parameter
- Discuss selection later

Selecting the Right Loss Function

- Depends on the problem type
- Always compare final output z_{Oi} with target y_i

Problem	Target y_i	Output z_{Oi}	Loss function	Formula
Regression	y_i = Scalar real	z_{Oi} = Prediction of y_i Scalar output / sample	Squared / MSE loss	$\sum_i (y_i - z_{Oi})^2$
Regression with vector samples	$\boldsymbol{y}_i = (y_{i1}, \dots, y_{iK})$	z_{Oik} = Prediction of y_{ik} K outputs / sample	Squared / MSE loss	$\sum_{ik} (y_{ik} - z_{Oik})^2$
Binary classification	$y_\mathrm{i} = \{0,1\}$	z_{Oi} = "logit" score Scalar output / sample	Binary cross entropy	$\sum_i [\ln(1 + e^{y_i z_i}) - y_i z_{Oi}]$
Multi-class classification	$y_\mathrm{i} = \{1, \dots, K\}$	z_{Oik} = "logit" scores K outputs / sample	Categorical cross entropy	$\sum_i \ln\left(\sum_k e^{z_{Oik}}\right) - \sum_k r_{ik} z_{Oik}$

Note on Indexing

- Neural networks are often processed in batches
  - Set of training or test samples
- Need different notation for single and batch input case
- For a single input $\boldsymbol{x}$
  - x_j = j-th feature of the input
  - $z_{H,j}, u_{H,j}, z_{O,j}$ = j-th component of hidden and output variables
  - H and O stand for Hidden and Output. Not an index
  - Write x, z_O, y if they are scalar (i.e. do not write index)
- For a batch of inputs $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$
  - x_{ij} = j-th feature of the input sample i
  - $z_{H,ij}, u_{H,ij}, z_{O,ij}$ = j-th component of hidden and output variables for sample i

Dimension Example

- Consider a neural network with:
  - $d = 5$ inputs, $N_H = 20$ hidden units
  - Output is for $K = 3$ class classification $\Rightarrow 3$ output units
- Dimensions for one input sample:
  - Input x: vector shape 5
  - Hidden units z_H, u_H: vector shape 20
  - Output units z_O, u_O: vector shape 3
- Dimensions for batch of 100 samples
  - Input $\boldsymbol{x}$: matrix shape (100,5)
  - Hidden units z_H, u_H: matrix shape (100,20)
  - Output units z_O, u_O: matrix shape (100,3)