Midterm Review

Autograd: Decomposing model into atomic operations, making a computational graph

ResNets: Know what a residual connection is and design considerations

Linear Regression

$$\hat{y} = \underbrace{w}_{\text{parameters}}^{T} \overset{\text{input}}{x} + \underbrace{b}$$

$$L = \frac{1}{2}(\hat{y} - \underbrace{y}_{\substack{\text{true}\\\text{value}}})^2$$

$$w \leftarrow w - \underbrace{\eta}_{\substack{\text{learning}\\\text{rate}}} \nabla_w L$$

$$\frac{dL}{dw} = \frac{dL}{d\hat{y}} \frac{d\hat{y}}{dw} = (\hat{y} - y)x$$

Softmax Regression

(logistic)

$$o = Wx + b$$

"scores"

$$\hat{y} = \text{softmax}(o) \qquad \hat{y}_j = \frac{\exp(o_j)}{\sum_k \exp(o_k)}$$

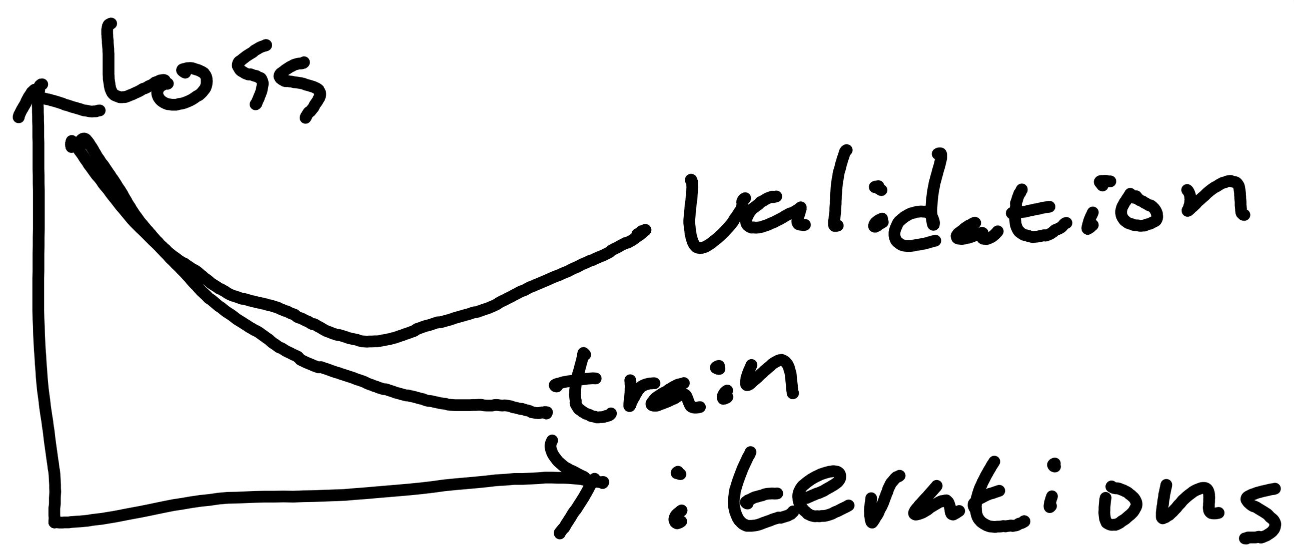
$$L = -\sum_j y_j \log \hat{y}_j = -\log \hat{y}_i$$

label (one-hot)

correct class

$$\frac{dL}{do_j} = \text{softmax}(o)_j - y_j$$

Overfitting and regularization



Weight decay:

add loss term $\lambda \|w\|^2$

↳ parameter vector

Dropout:

$$h' = \begin{cases} 0 & \text{with probability } p \\ \frac{h}{1-p} & \text{with probability } 1-p \end{cases}$$

Adaptive gradient descent methods

Momentum:

$$v_t \leftarrow \beta v_{t-1} + g_t$$

gradient at iteration t

$\nabla_\theta L$

$$\theta_t \leftarrow \theta_{t-1} - \eta v_t$$

Adam:

(complex update)

$$g'_t = \frac{\eta \hat{v}_t}{\sqrt{\hat{s}_t} + \varepsilon}$$

MLP

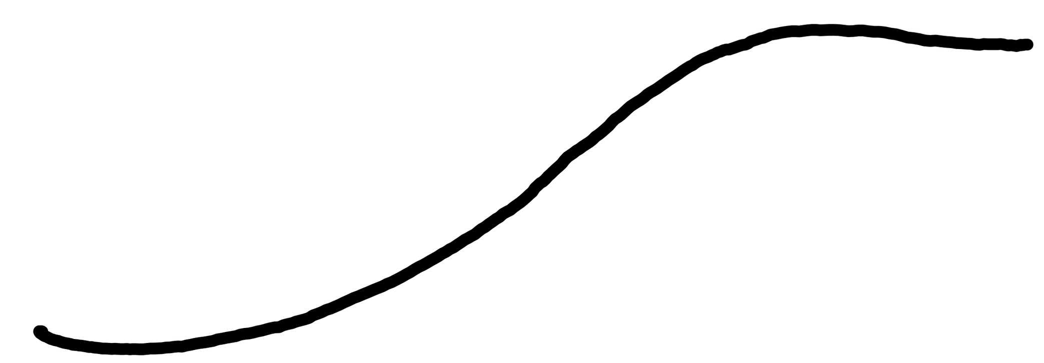
feed forward, dense, fully-connection, "DNN"

$$h = \phi(W_1 x + b_1)$$

hidden rep. non-linearity

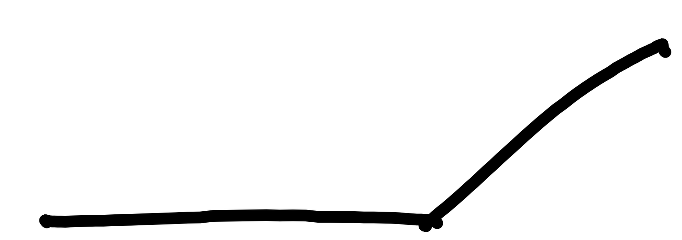
$$o = W_2 h + b_2$$

$$\text{sigmoid}(x) = \frac{1}{1+\exp(-x)}$$



$$\frac{d\,\text{sigmoid}}{dx} = \text{sigmoid}(x)(1-\text{sigmoid}(x))$$

$$\text{ReLU}(x) = \max(0, x)$$



$$\frac{d\,\text{ReLU}}{dx} = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases}$$

Backpropagation

$$z_m = W_m a_{m-1} + b_m$$

$$a_m = \phi_m(z_m)$$

C = cost function of a_L and y

$$\frac{dC}{dW_m} = \frac{dC}{dz_m} a_{m-1}^T$$

$$\frac{dC}{dz_m} = \underbrace{\left(W_{m+1}^T \frac{dC}{dz_{m+1}} \right)}_{\text{recursion!}} \circ \frac{da_m}{dz_m}$$

Initialization

- Assume $x \sim \mathcal{N}(0, I)$
- Assume initial $w \sim \mathcal{N}(0, \sigma^2 I)$
- Ignore nonlinearities

$$\sigma = \sqrt{\frac{2}{n_{in} + n_{out}}}$$

Autograd

$$h = \mathrm{ReLU}(W_h x + b_h)$$

$$o = W_o h + b_o$$

$$L = (y - o)^2$$

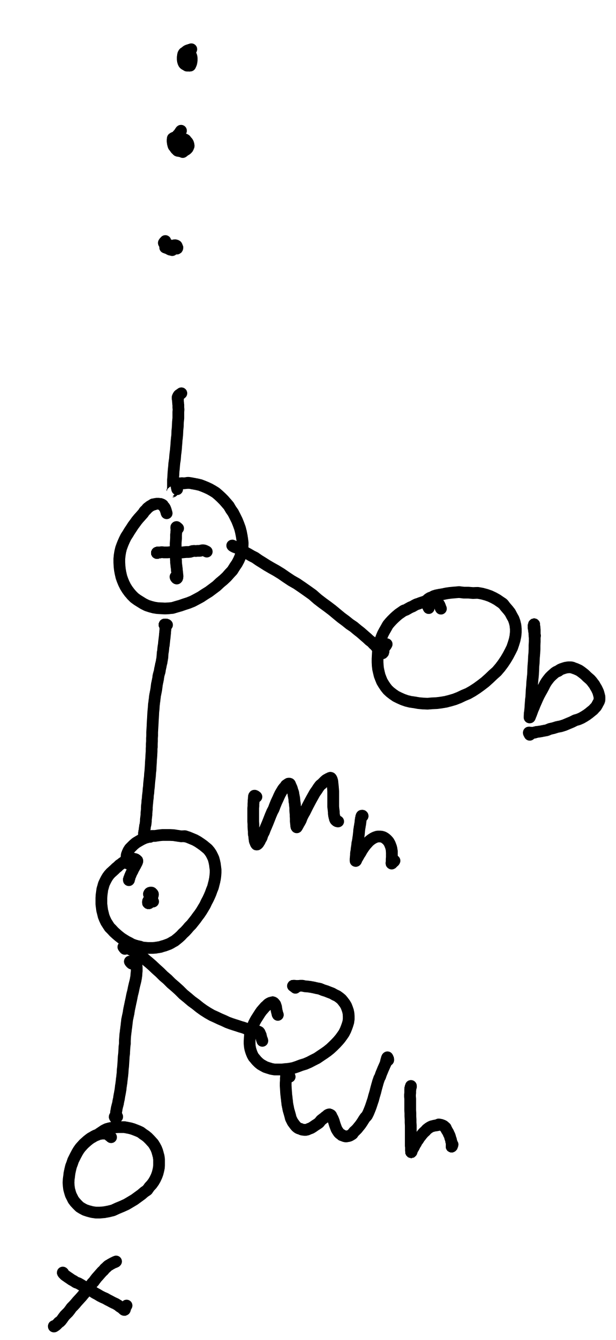
$$m_h = W_h x$$

$$z_h = m_h + b_h$$

$$h = \mathrm{ReLU}(z_h)$$

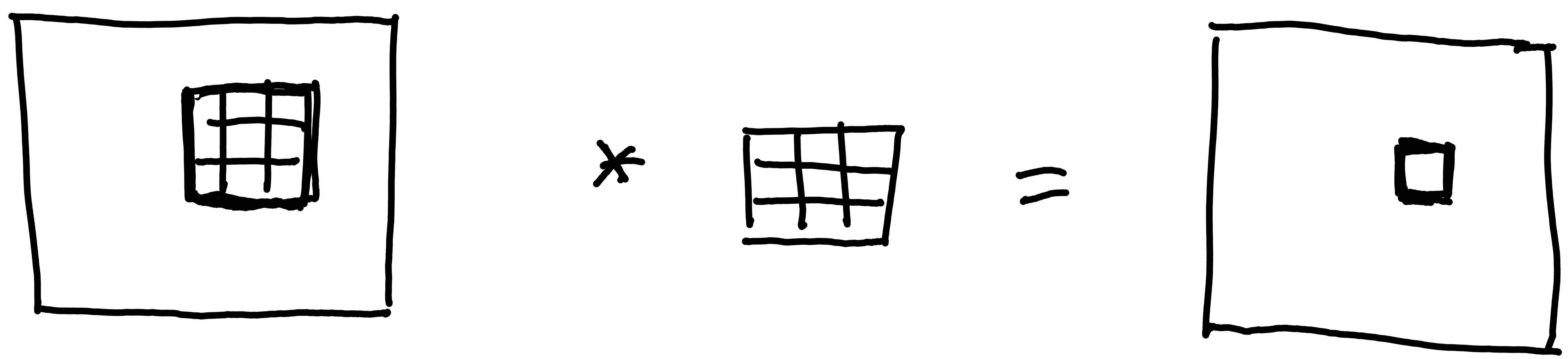
$$\vdots$$

$$\frac{dL}{dW_h} = \frac{dL}{de} \frac{de}{do} \frac{do}{m_o} \cdots$$



Convolution

$$H_{ijd} = \sum_{a=-\Delta}^{\Delta} \sum_{b=-\Delta}^{\Delta} \sum_{c} V_{abcd} \, X_{i+a,\, i+b,\, c}$$



Other factors:

- padding
- striding
- multiple input and output channels
- receptive field

ConvNet ingredients

- Convolutions
- Pooling - apply a reduction over a region of the input
- Dense layers

Batch Norm

$$\text{BatchNorm}(x) = \gamma \odot \frac{x - \hat{\mu}_B}{\hat{\sigma}_B + \varepsilon} + \beta$$

$$\hat{\mu}_B = \frac{1}{|B|} \sum_{x \in B} x$$

$$\hat{\sigma}_B^2 = \frac{1}{|B|} \sum_{x \in B} (x - \hat{\mu}_B)^2$$

γ, β learnable params, same shape as x

ResNets

Residual connection: $f(x) + x$

(optionally $\underset{1\times 1 \text{ conv}}{\text{process}}$ x to match shape)