Fitting Neural Networks

cost \((\hat{y} - y)^T (\hat{y} - y)\), or \(-\log P(y \mid \hat{y})\)

\[
\begin{align*}
\hat{y} &= g^{(0)}(a^{(0)}) \\
\hat{y} &= g^{(0)}(a^{(0)}) \\
\hat{y} &= g^{(0)}(a^{(0)}) \\
\hat{y} &= g^{(0)}(a^{(0)}) \\
\hat{y} &= g^{(0)}(a^{(0)}) \\
\hat{y} &= g^{(0)}(a^{(0)})
\end{align*}
\]

\(a^{(0)} = \left[ \begin{array}{c}
\text{vec}(W^{(1)}) \\
\text{vec}(b^{(1)}) \\
\text{vec}(W^{(2)}) \\
\vdots \\
\text{vec}(b^{(2)})
\end{array} \right]
\)

\(\theta \leftarrow \theta - \gamma \nabla_\theta c\)

**Today:** Termination (early stopping)

Computing \(\nabla_\theta c\) (reverse-mode diff.)
Early Stopping

Form regularization, alternative to L2 or do as well

Cost/c

Training cost

Validation cost
not "overfitting"

overfitting

val error - train error

keep the model at this iteration

Iteration #
Every sweep through training set "an epoch":

If val. cost is the smallest we've seen
Store weights $\theta$ & val. cost

If val. cost hasn't improved in 20 evaluations of val. set:
Stop
return best $\theta$

reduce learning rate $\eta$

$\eta_t$ # iterations, $t$

Last update
Reverse-mode differentiation "Back-propagation"

Works on a compute graph (DAG)

Directed acyclic

Strategy

For every intermediate \( z \)

get \( \bar{z} = \frac{\partial c}{\partial z} \)

For a matrix \( \bar{z}_{ij} = \frac{\partial c}{\partial z_{ij}} \)

Start at end of computation

Example: \( c = (f-y)^2 \)

\[ \bar{f} = \frac{\partial c}{\partial f} = 2(f-y) \]

If have \( \bar{f} \) and \( y \)

\[ c = \sum_k (f_k-y_k)^2 \]

\[ \bar{f}_{ij} = \frac{\partial c}{\partial f_{ij}} = 2(f_i-y_i) \]

\[ \bar{f} = 2(f-y) \]
We combine local propagation rules

\[ \cdots \rightarrow u \rightarrow \cdots \rightarrow c \]

Local function: \( w = \frac{u}{v} \)

Assume we have \( \bar{w} = \frac{\partial c}{\partial w} \)

Want \( \bar{u} = \frac{\partial c}{\partial u} \), \( \bar{v} = \frac{\partial c}{\partial v} \)

Chain rule

\[ \bar{u} = \frac{\partial c}{\partial u} = \frac{\partial c}{\partial w} \frac{\partial w}{\partial u} \]

\[ \bar{v} = \frac{\partial c}{\partial v} = \frac{\partial c}{\partial w} \frac{\partial w}{\partial v} \]

Derivative of local function

Depends on \( u \) and \( v \) and/or \( w \)

In example: \( \frac{1}{v} \)

In example: \( \frac{-u}{v^2} \)

2019 L21 ⑤
Example: Matrix Multiplication

NXN matrix \( N^2 \) numbers in it

\( O(\cdot) \) cost of matrix-matrix

\[
N^3 = N^2 \log N^7 \Rightarrow N^{2.373}
\]

\[
\rightarrow X \xrightarrow{\otimes} Z \rightarrow \ldots \rightarrow c
\]

\[
z = xy
\]

\[
\ldots \leftarrow \bar{x} \leftarrow \bar{z} \leftarrow \ldots \leftarrow \bar{s} \leftarrow c
\]

\[
X_j = \frac{\partial c}{\partial x_{ij}} = \sum_{m,n} \frac{\partial c}{\partial z_{mn}} \frac{\partial z_{mn}}{\partial x_{ij}} = \sum_n \bar{z}_{in} \bar{y}_{jn} (Y_h^T)_{ij}
\]

\[
\bar{x} = \bar{z} Y^T
\]

\[
\bar{y} = \bar{x}^T \bar{z}
\]
\[ z_{mn} = \sum_{p} x_{mp} y_{pn} \]
\[ = x_{m1} y_{1n} + x_{m2} y_{2n} + \ldots + x_{mj} y_{jn} + \ldots \]

\[ z = X Y \]
\[ M \times N \quad M \times P \quad P \times N \]

\[ \bar{Y} = X^T \bar{z} \]
\[ P \times N \quad P \times M \quad M \times N \]