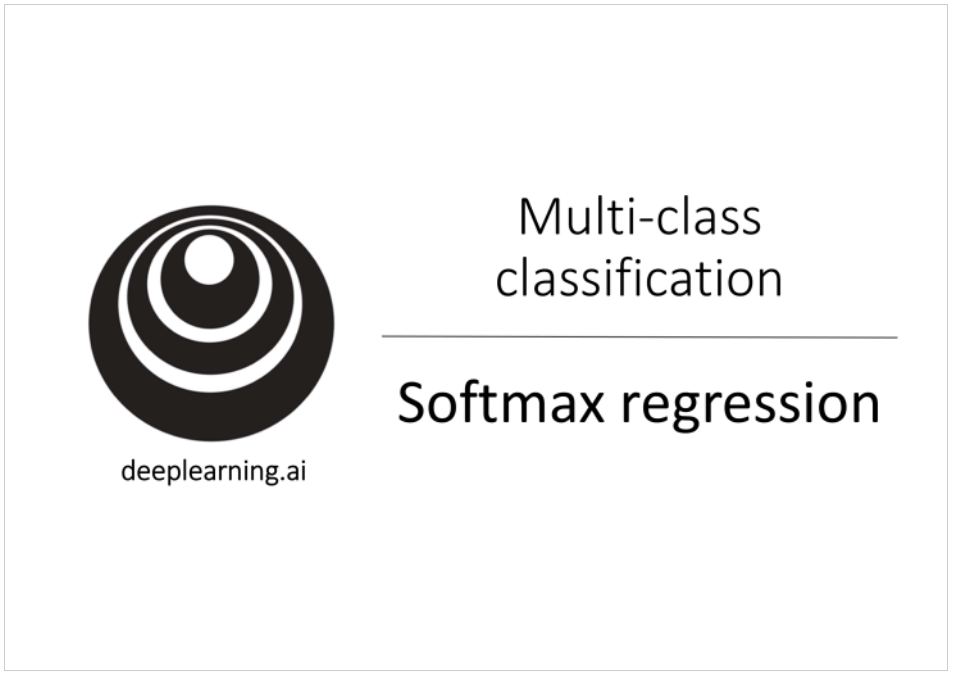
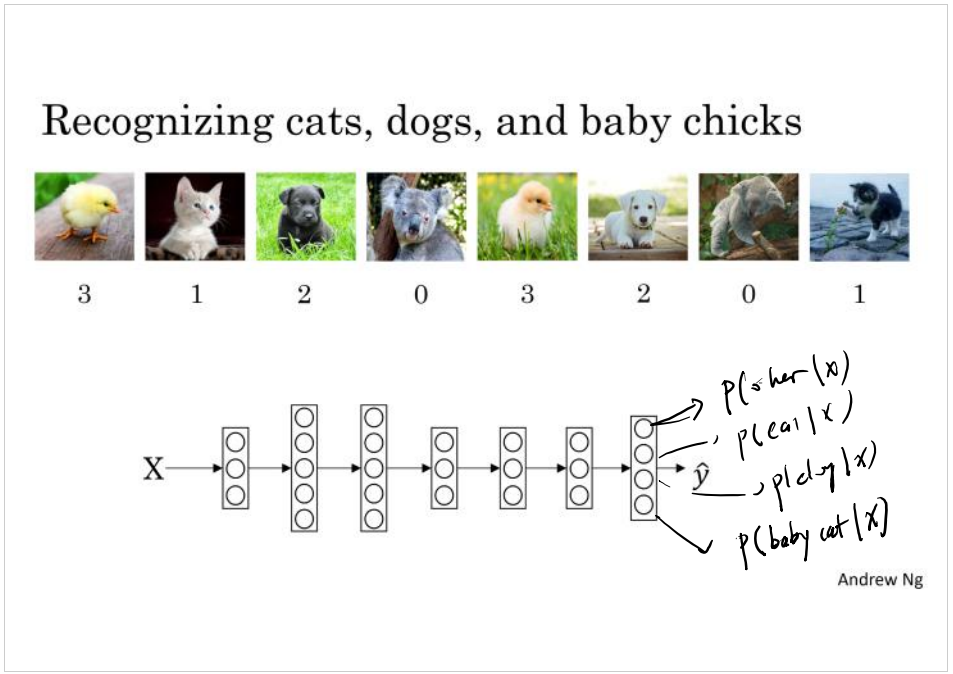
Multi-class classification

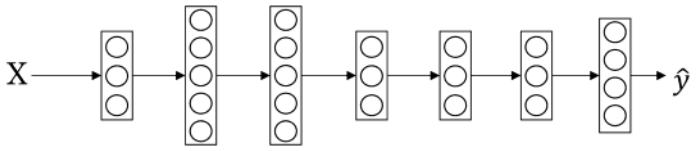
Monday, June 4, 2018 9:32 PM

multiclass1





Softmax layer



$z^{[L]} = W^{[L]} a^{[L-1]} + b^{[L]}$

Activation function

$t = e^{z^{[L]}}$

$a^{[L]} = \frac{t_i}{\sum_{j=1}^{4} t_j}$ $t = e^{z^{[L]}}$

(4, 1)

$z^{[L]} = \begin{bmatrix} 5 \\ 2 \\ -1 \\ 3 \end{bmatrix}$

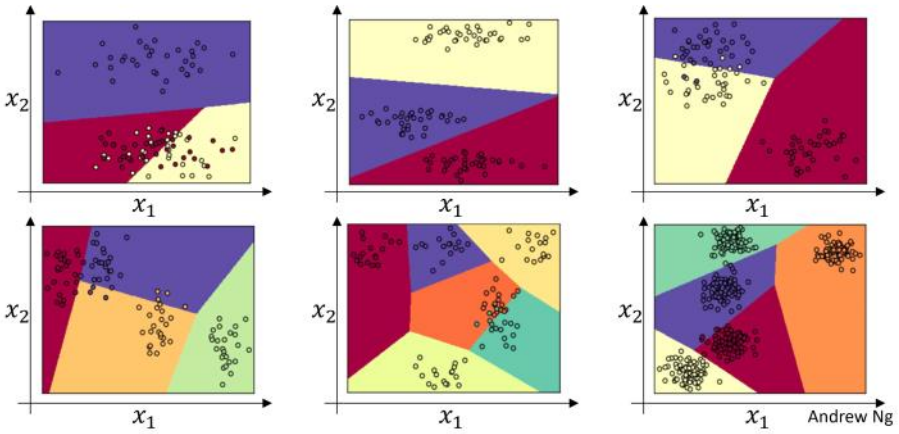
$t = \begin{bmatrix} e^5 \\ e^2 \\ e^{-1} \\ e^3 \end{bmatrix} = \begin{bmatrix} 148.4 \\ 7.4 \\ 0.4 \\ 20.1 \end{bmatrix}$

$\sum_{j=1}^{4} t_j = 176.3$

$a^{[L]} = \frac{t}{176.3}$

Andrew Ng

Softmax examples



What's common: Decision boundary between any two classes are LINEAR.

when no hidden layer.

Andrew Ng

multiclass2

deeplearning.ai

Multi-class classification

Trying a softmax classifier

Understanding softmax

$z^{[L]} = \begin{bmatrix} 5 \\ 2 \\ -1 \\ 3 \end{bmatrix}$ $t = \begin{bmatrix} e^5 \\ e^2 \\ e^{-1} \\ e^3 \end{bmatrix}$

$a^{[L]} = g^{[L]}(z^{[L]}) = \begin{bmatrix} e^5/(e^5+e^2+e^{-1}+e^3) \\ e^2/t^T t \\ \vdots \end{bmatrix}$

softmax $\Longleftrightarrow$ HARDMAX $a^{[L]} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$ set the maxima as 1

Andrew Ng

When #Class = 2, then it is just a logistic regression

Loss function

Define the loss function

$y = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$ ← cat. $\hat{y} = \begin{bmatrix} 0.3 \\ 0.2 \\ 0.1 \\ 0.4 \end{bmatrix}$ ← the output.

其实是 $\hat{p}_1^{y_1} \hat{p}_2^{y_2} \hat{p}_3^{y_3} \hat{p}_4^{y_4}$

$\mathcal{L}(\hat{y}, y) = -\sum_{j=1}^{4} y_j \log \hat{y}_j$

$= -y_2 \log \hat{y}_2 = -\log \hat{y}_2$ => so the algorithm makes $\hat{y}_2$ big.

$J(w^{[1]}, b^{[1]}, \ldots) = \frac{1}{m} \sum_{i=1}^{m} \mathcal{L}(\hat{y}^{(i)}, y^{(i)})$ ← 全部数据集上 loss 加总.

$Y = [y^{(1)}, y^{(2)} \ldots y^{(m)}] = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots \\ 1 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 1 & \\ 0 & 0 & 1 & 0 & \end{bmatrix}$

$\hat{Y} = [\hat{y}^{(1)} \ldots \hat{y}^{(m)}]$

$= -y_2 \log \hat{y}_2 = -\log \hat{y}_2$ => So the algorithm makes $\hat{y}_2$ big.

Andrew Ng

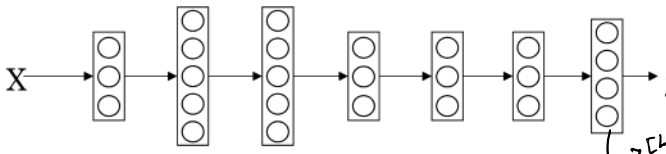
this is just MLE

$Y = [y^{(1)} \dots y^{(m)}]$ $\begin{bmatrix} 0 & 0 & 0 & 1 & \dots \\ 0 & 0 & 1 & 0 & \end{bmatrix}$

$\hat{Y} = [\hat{y}^{(1)} \dots \hat{y}^{(m)}]$

$= \begin{bmatrix} 0.3 & \dots \\ 0.2 & \\ 0.1 & \\ 0.4 & \end{bmatrix}$

Summary of softmax classifier



$(z^{[L]} \to a^{[L]} = \hat{y} = \mathcal{L}(\hat{y}, y)$

$(4,1)$

Backprop

$dz^{[L]} = \hat{y} - y$ How to derive?

using tf. ~~for~~ only need to do forward prop right.

$\frac{\partial J}{\partial z^{[L]}}$

查查这个 backprop 怎么推

Andrew Ng