Learning from multiple tasks

Multi-task learning

deeplearning.ai

Simplified autonomous driving example

$X^{(i)}$

Pedestrians

Cars

Stop signs

Traffic lights
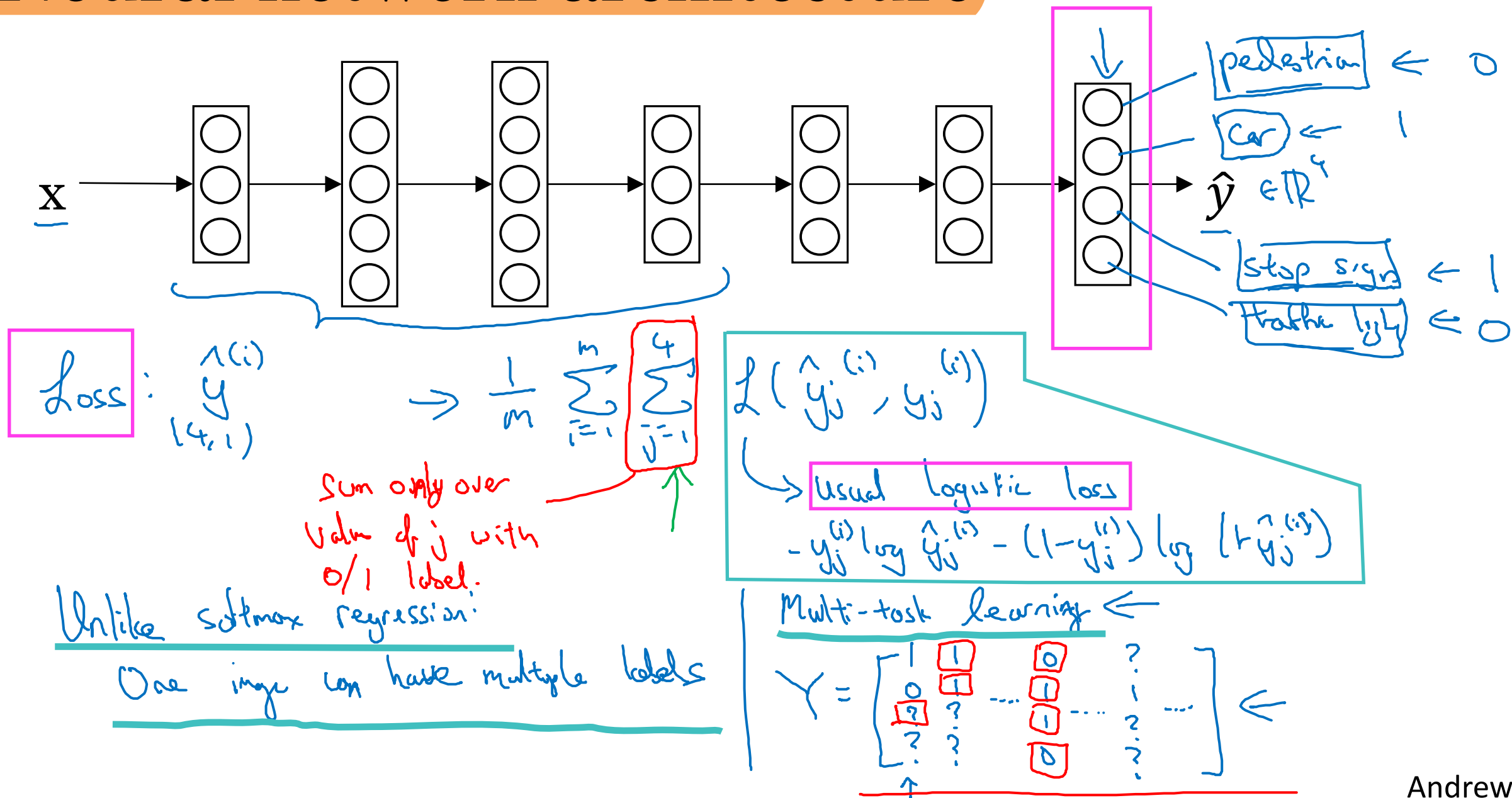
$\vdots$

$y^{(i)}$ $(4,1)$

$\begin{array}{c} 0 \\ 1 \\ 1 \\ 0 \\ \vdots \end{array}$

$$Y = \begin{bmatrix} y^{(1)} & y^{(2)} & y^{(3)} & \cdots , & y^{(m)} \end{bmatrix}$$

$(4, m)$

Output

Neural network architecture

x → [network layers] → $\hat{y}$

pedestrian ← 0

Car ← 1

$\in \mathbb{R}^4$

Stop sign ← 1

traffic light ← 0

Loss:

$\hat{y}^{(i)}$

y (4,1)

$\rightarrow \frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{4}$

Sum only over
value of j with
0/1 label.

$L(\hat{y}_j^{(i)}, y_j^{(i)})$

↳ Usual logistic loss

$-y_j^{(i)} \log \hat{y}_j^{(i)} - (1 - y_j^{(i)}) \log(1 - \hat{y}_j^{(i)})$

Multi-task learning ←

Unlike softmax regression:

One image can have multiple labels

$$Y = \begin{bmatrix} 1 & 0 & ? \\ 1 & 0 & ? \\ 0 & ? & 1 & ? \\ ? & ? & 1 & ? \\ ? & ? & 0 & ? \end{bmatrix}$$

Andrew Ng
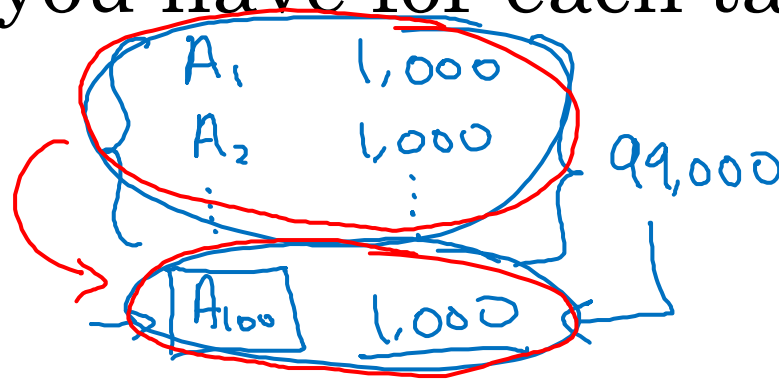
When multi-task learning makes sense

- Training on a set of tasks that could benefit from having shared lower-level features.

- Usually: Amount of data you have for each task is quite similar.

$A \quad 1,000,000$

$\downarrow \qquad \downarrow$

$B \quad 1,000$

$A_1 \qquad 1,000$
$A_2 \qquad 1,000$
$\vdots \qquad \vdots \qquad 99,000$

$A_{100} \qquad 1,000$

- Can train a big enough neural network to do well on all the tasks.