deeplearning.ai

# Regularizing your neural network

## Regularization

# Logistic regression

$$w \in \mathbb{R}^{n_x}, \quad b \in \mathbb{R}$$

$$\lambda = \text{regularization parameter}$$

$$\text{lambda} \qquad \text{lambd}$$

$$\min_{w,b} J(w,b)$$

$$J(w,b) = \frac{1}{m} \sum_{i=1}^{m} \mathcal{L}\left(\hat{y}^{(i)}, y^{(i)}\right) + \frac{\lambda}{2m} \|w\|_2^2 \qquad + \frac{\lambda}{2m} b^2$$

$$\text{omit}$$

$L_2$ regularization

$$\|w\|_2^2 = \sum_{j=1}^{n_x} w_j^2 = w^T w \quad \longleftarrow$$

$L_1$ regularization

$$\frac{\lambda}{2m} \sum_{j=1}^{n_x} |w_j| = \frac{\lambda}{2m} \|w\|_1$$

$w$ will be sparse

# Neural network

$$\rightarrow J(w^{[1]}, b^{[1]}, \ldots, w^{[L]}, b^{[L]}) = \frac{1}{m} \sum_{i=1}^{m} \mathcal{L}(\hat{y}^{(i)}, y^{(i)}) + \frac{\lambda}{2m} \sum_{l=1}^{L} \|w^{[l]}\|_F^2$$

$$\|w^{[l]}\|_F^2 = \sum_{i=1}^{n^{[l]}} \sum_{j=1}^{n^{[l-1]}} (w_{ij}^{[l]})^2$$

$$w^{[l]} : (n^{[l]}, n^{[l-1]})$$

"Frobenius norm"

$$\|\cdot\|_2^2 \qquad \|\cdot\|_F^2$$

$$dw^{[l]} = \boxed{(\text{from backprop}) + \frac{\lambda}{m} w^{[l]}}$$

$$\frac{\partial J}{\partial w^{[l]}} = dw^{[l]}$$

$$\rightarrow w^{[l]} := w^{[l]} - \alpha \, dw^{[l]}$$

"Weight decay"

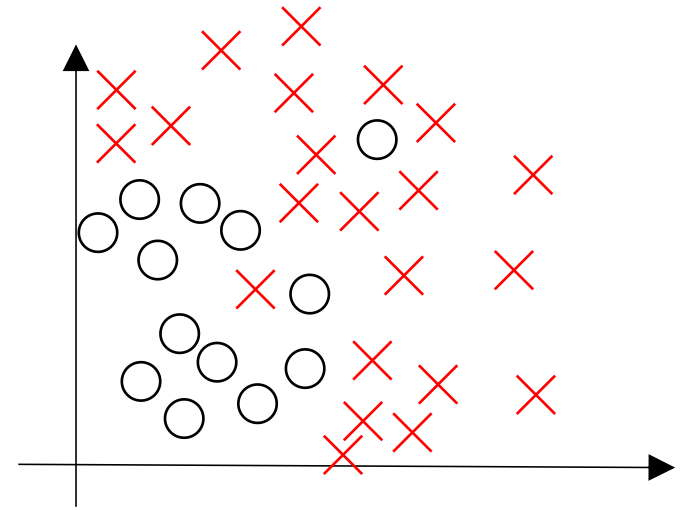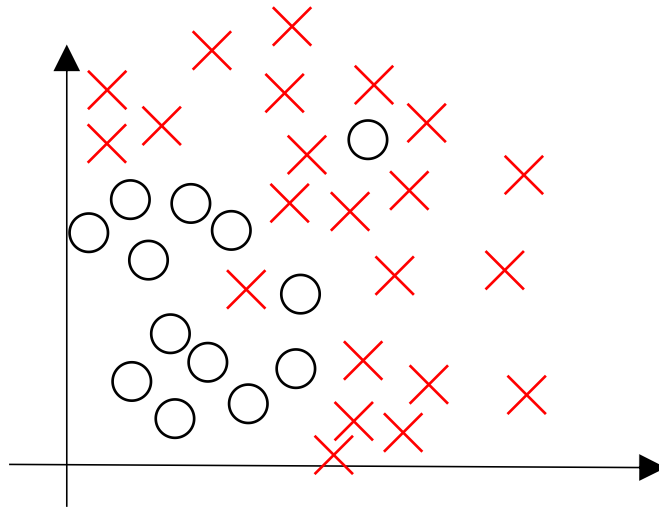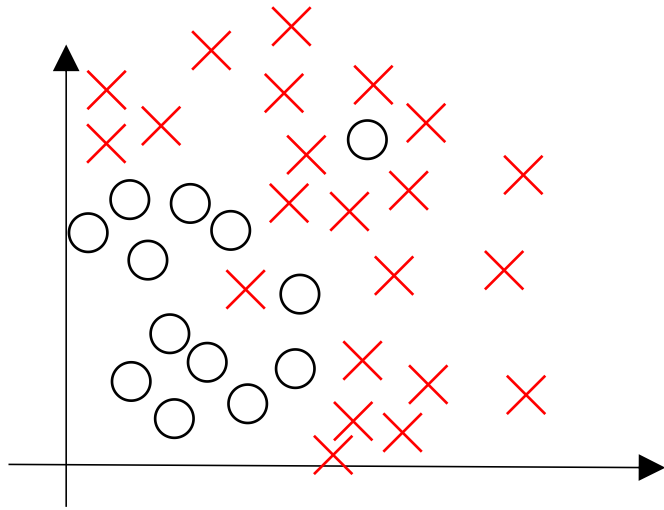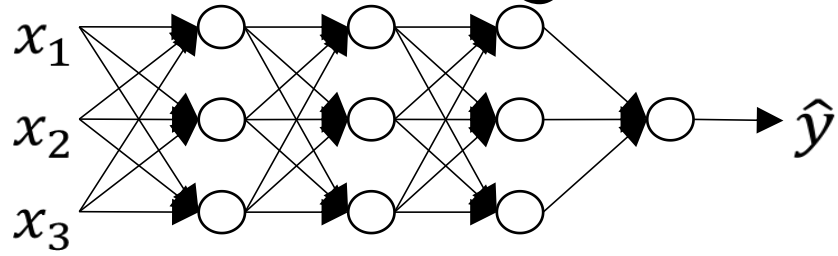$$w^{[l]} := w^{[l]} - \alpha \left[ (\text{from backprop}) + \frac{\lambda}{m} w^{[l]} \right]$$

$$= w^{[l]} - \frac{\alpha \lambda}{m} w^{[l]} - \alpha (\text{from backprop})$$

$$= \underbrace{\left(1 - \frac{\alpha \lambda}{m}\right)}_{<1} \underline{w^{[l]}} - \alpha \underline{(\text{from backprop})}$$

Andrew Ng

# Neural network

$$J(w^{[1]}, b^{[1]}, \ldots, w^{[L]}, b^{[L]}) = \frac{1}{m} \sum_{i=1}^{m} \mathcal{l}(\hat{y}^{(i)}, y^{(i)}) + \frac{\lambda}{2m} \sum_{l=1}^{L} \|w^{[l]}\|^2$$

# How does regularization prevent overfitting?

# How does regularization prevent overfitting?