Regularizing your neural network

Regularization

deeplearning.ai

Logistic regression

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

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

$\lambda = $ regularization parameter

lambda   lambd

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

$$+ \frac{\lambda}{2m} b^2 \quad \text{Omit}$$

L_2 regularization

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

L_1 regularization

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

w will be sparse

Andrew Ng

Neural network

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

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

"Frobenius norm" $\qquad \|\cdot\|_2^2 \qquad \|\cdot\|_F^2$

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

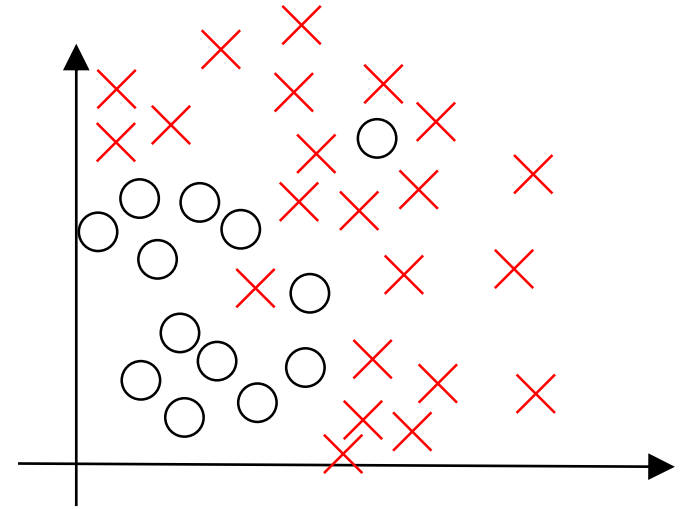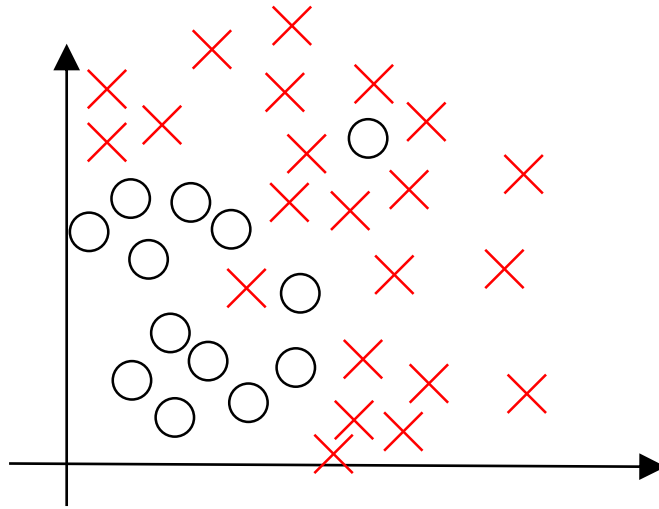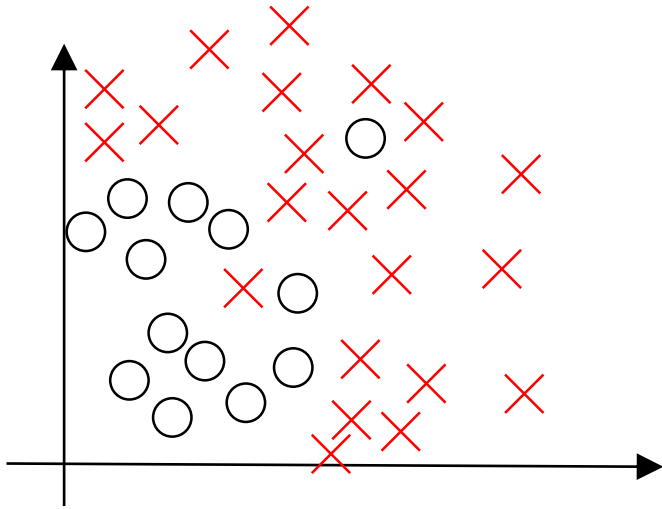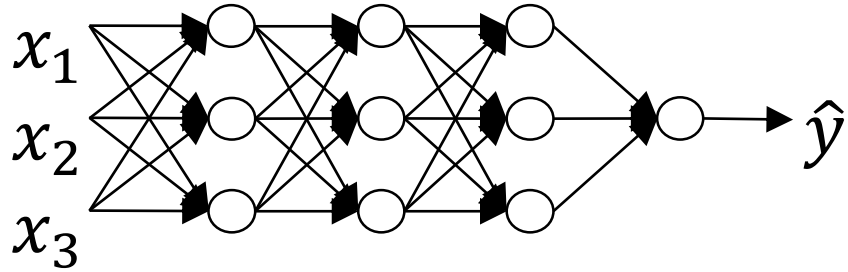$$\rightarrow w^{[l]} := w^{[l]} - \alpha \, dw^{[l]}$$

"Weight decay"

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

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

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

Andrew Ng

How does regularization prevent overfitting?

How does regularization prevent overfitting?