Índice

La neurona logística

- ¿En qué se basaba el algoritmo de aprendizaje del perceptrón?
  - En ir modificando los parámetros según fallamos la predicción
- Ahora debemos cambiar la filosofía
  - Modificar los parámetros de tal manera que la salida de la neurona se parezca cada vez más a lo que debería obtener en realidad (coste)

La neurona logística

- Modificación sobre el perceptrón
  - Modificamos la función de activación de la neurona
    - Evitamos la función escalón por una función "suave" (sigmoide)

$$h_\theta(x) = \sigma(\theta_0 x_0 + \cdots + \theta_n x_n)$$

  - Definimos una función de coste que queremos minimizar
    - Evitamos el problema de la convergencia

La neurona logística

- Objetivo
  - Dado $\{(x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)})\}$, queremos $\hat{y}^{(i)} = y^{(i)}\ para\ todo\ i$
- ¿Cómo medimos lo alejados que están $\hat{y}^{(i)}$ de $y^{(i)}$?

$$L(\hat{y}^{(i)}, y^{(i)}) = -(y^{(i)} \log \hat{y}^{(i)} + (1-y) \log(1-\hat{y}^{(i)}))$$

- ¿Cómo definir una función de coste?
  - Necesitamos encontrar los parámetros w, b tales que minimicemos $L(\hat{y}^{(i)} - y^{(i)})$ para todo i

La neurona logística

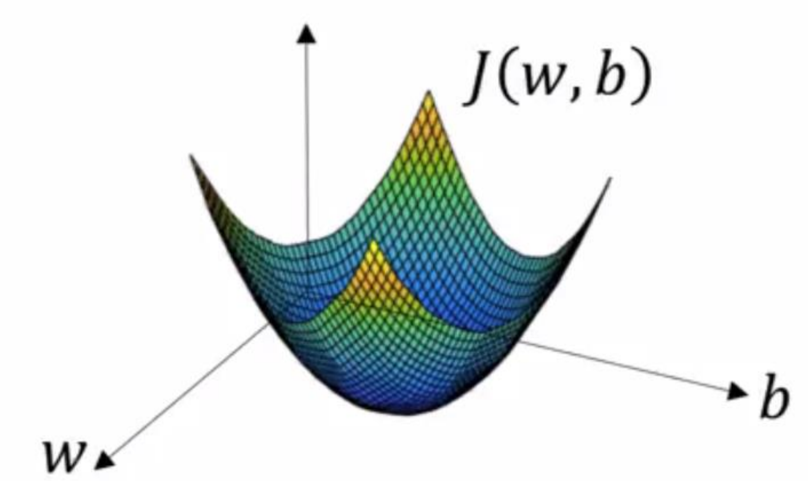
- Definimos una función de coste $J: R^{n+1} \to R^{+}$ dada por

$$J(w,b) = \frac{1}{m}\sum_{i=1}^{m} L\left(\hat{y}^{(i)} - y^{(i)}\right) =$$

$$-\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log(\hat{y}^{(i)}) + \left(1 - y^{(i)}\right)\log(1 - \hat{y}^{(i)})\right]$$

La neurona logística

- Objetivo
  - Encontrar los mejores valores de w y b



La neurona logística