Vectorization

$z = w^T x + b$, $w = \begin{bmatrix} \vdots \\ \vdots \end{bmatrix}$, $x = \begin{bmatrix} \vdots \\ \vdots \end{bmatrix}$ $\quad w \in \mathbb{R}^n$ $\quad x \in \mathbb{R}^n$

<Non-Vector>

```
z = 0
for i in range(n):
    z = z + w[i] * x[i]
    z = z + b
```

→ "벡터화"

```
z = np.dot(w, x) + b
```

내적

⇒ 벡터화를 할 경우
코드 간결 ⊕ 빠른 연산 ⊕ 시간 단축.

Vectorizing Logistic Regression

<Vectorizing Logistic Regression>

$z^{(1)} = w^T x^{(1)} + b$ $\quad a^{(1)} = \sigma(z^{(1)})$

$z^{(2)} = w^T x^{(2)} + b$ $\quad a^{(2)} = \sigma(z^{(2)})$

$z^{(3)} = w^T x^{(3)} + b$ $\quad a^{(3)} = \sigma(z^{(3)})$

$X = \begin{bmatrix} \vdots & \vdots & & \vdots \\ x^{(1)} & x^{(2)} & \cdots & x^{(m)} \\ \vdots & \vdots & & \vdots \end{bmatrix} \in \mathbb{R}^{n \times m}$

b는 실제로 1x1 이지만, 연산에서 1xm인 것처럼 계산! ⇒ "브로드 캐스팅"

$Z = [z^{(1)} z^{(2)} \cdots z^{(m)}] = w^T X + b = [w^T x^{(1)} + b, \; w^T x^{(2)} + b, \; \cdots, \; w^T x^{(m)} + b]$

$= np.dot(w^T, x) + b$

"내적"

<Implementing Logistic Regression>

$J = 0$, $dw_1 = 0$, $dw_2 = 0$, $db = 0$ → $dw = np.zeros((n,1))$

for i = 1 to m:

$z^{(i)} = w^T x^{(i)} + b$

$a^{(i)} = \sigma(z^{(i)})$

$J \mathrel{+}= -[y^{(i)} \log a^{(i)} + (1-y^{(i)}) \log(1-a^{(i)})]$

$dz^{(i)} = a^{(i)}(1-a^{(i)})$

$dw_1 \mathrel{+}= x_1^{(i)} dz^{(i)}$

$dw_2 \mathrel{+}= x_2^{(i)} dz^{(i)}$ → $dw = dw + x^{(i)} \cdot dz^{(i)}$

$db \mathrel{+}= dz^{(i)}$

$J = J/m$, $dw_1 = dw_1/m$, $dw_2 = dw_2/m$, $db = db/m$ → $dw = dw/m$

Vectorization을 통해 for문 제거하기

$Z = w^T X + b$

$= np.dot(w^T, X) + b$

$A = \sigma(Z)$

$dZ = A - Y$

$dw = \frac{1}{m} np.dot(X, dZ^T)$

$db = \frac{1}{m} np.sum(dZ)$

$w := w - \alpha dw$

$b := b - \alpha db$

learning rate

Logistic Regression cost function

If $y=1$, $p(y|x) = \hat{y}$ ← 그럴때 y=1일 확률

If $y=0$, $p(y|x) = 1-\hat{y}$ ← 〃 y=0 〃

↓ 두 식을 합침

$$p(y|x) = \hat{y}^{y}(1-\hat{y})^{(1-y)} \Leftarrow \begin{cases} \text{If } y=1, & p(y|x) = \hat{y}(1-\hat{y})^{0} = \hat{y} \\ \text{If } y=0, & p(y|x) = \hat{y}^{0}(1-\hat{y})^{1} = 1-\hat{y} \end{cases}$$

* 로그함수의 성질과 관하여 위식은...

$\log(p(y|x)) = \log(\hat{y}^{y}(1-\hat{y})^{(1-y)})$ 와 동일.

우리의 목표: 확률 maximize $= \log(\hat{y}^{y}(1-\hat{y})^{(1-y)})$ → ∴ $-\log(p(y|x))$를 minimize

손실함수: $\mathcal{L}(\hat{y}, y) = -\log(p(y|x)) = -(\hat{y}^{y}(1-\hat{y})^{(1-y)})$

비용함수: 손실함수들의 "평균"

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

Broad Casting

[EX)

$$\begin{bmatrix}1\\2\\3\\4\end{bmatrix} + 100 = \begin{bmatrix}101\\102\\103\\104\end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix}100\\100\\100\\100\end{bmatrix}$$

,

$$\begin{bmatrix}1 & 2 & 3\\4 & 5 & 6\end{bmatrix} + \begin{bmatrix}100 & 200 & 300\end{bmatrix} = \begin{bmatrix}101 & 202 & 303\\104 & 205 & 306\end{bmatrix}$$

(2×3) (1×3)

$$\Downarrow$$

$$\begin{bmatrix}100 & 200 & 300\\100 & 200 & 300\end{bmatrix}$$

<General principle>

① $(m \times n)$ $\pm$ $*$ $/$ $(1 \times n)$ → m번 반복 (행)

$(m \times n)$ → n번 반복 (열)

② $(m \times n)$ $\pm$ $*$ $/$ 실수 (1×1) → $m \times n$ 번 반복 (모든 원소)

Quiz (헷갈렸던 문제 + 실수했던 문제 오답 노트)

- np.random
  - randint(n) : 0 ~ n-1 범위의 random한 정수 반환
  - rand(m,n) : 0 ~ 1 사이 난수 matrix array (m×n) 반환
  - randn(m,n) : Gaussian 분포를 따르는 난수 matrix array (m×n) 반환

- import numpy as np.

```
a = np.array([[3,0,1],[2,1,2]])
b = np.zeros([2,3])
c = np.ones([1,3])
d = a + b + c
print(d[1, 1:3])
```

$a = \begin{bmatrix} 3 & 0 & 1 \\ 2 & 1 & 2 \end{bmatrix}$, $b = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$, $c = [1, 1, 1]$

$a+b = \begin{bmatrix} 3 & 0 & 1 \\ 2 & 1 & 2 \end{bmatrix}$

$a+b+c = \begin{bmatrix} 4 & 0 & 2 \\ 3 & 2 & 3 \end{bmatrix}$

∴ d[1, 1:3]

⇒ 1번째 행에서 1~2번째 열까지 요소 추출 (index는 0부터)

∴ d = [2, 3]

np.dot(a, b) = a * b