Basics of Neural Network Programming

deeplearning.ai

Vectorization

Vectorization is basically the art of getting rid of explicit for loops in your code. I think the ability to perform vectorization has become a key skill

What is vectorization?

$$z = \underbrace{w^T x}_{} + b$$

$w = \begin{bmatrix} \vdots \\ \vdots \end{bmatrix}$ $x = \begin{bmatrix} \vdots \\ \vdots \end{bmatrix}$ $w \in \mathbb{R}^{n_x}$ $x \in \mathbb{R}^{n_x}$

Non-vectorized:

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

Vectorized

$$z = \underbrace{np.\underline{dot}(w, x)}_{w^T x} + b$$

parallelization instructions

→ GPU
→ CPU

SIMD - single instruction multiple data.

If you use built-in functions such as this np.function or other functions that don't require you explicitly implementing a for loop.
It enables Python numpy to take much better advantage of parallelism to do your computations much faster.
And this is true both computations on CPUs and computations on GPUs.

Andrew Ng

deeplearning.ai

Basics of Neural Network Programming

More vectorization examples

Neural network programming guideline

Whenever possible, avoid explicit for-loops.

Neural network programming guideline

Whenever possible, avoid explicit for-loops.

$u = Av$

$u_i = \sum_i \sum_j A_{ij} v_j$

```
u = np.zeros((n,1))
for i ...
    for j ...
        u[i] += A[i][j] * v[j]
```

```
u = np.dot(A,v)
```

Andrew Ng

Vectors and matrix valued functions

Say you need to apply the exponential operation on every element of a matrix/vector.

$$v = \begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix} \rightarrow u = \begin{bmatrix} e^{v_1} \\ e^{v_2} \\ \vdots \\ e^{v_n} \end{bmatrix}$$

```
u = np.zeros((n,1))
for i in range(n):
    u[i]=math.exp(v[i])
```

```
import numpy as np
u = np.exp(v)

np.log(v)
np.abs(v)
np.maximum(v, 0)
v**2        1/v
```

Logistic regression derivatives

$J = 0$, $\boxed{\cancel{dw1 = 0}, \cancel{dw2 = 0}}$, $db = 0$

$dw = np.zeros((n\_x, 1))$

```
for i = 1 to n:
```

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

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

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

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

for $j = 1 \ldots n_x$

$dw_j \mathrel{+}= \ldots$

$\boxed{\begin{aligned} &\cancel{dw_1 \mathrel{+}= x_1^{(i)} dz^{(i)}} \\ &\cancel{dw_2 \mathrel{+}= x_2^{(i)} dz^{(i)}} \end{aligned}}$

$n_x = 2$

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

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

$J = J/m$, $\boxed{\cancel{dw_1 = dw_1/m}, \cancel{dw_2 = dw_2/m}}$, $db = db/m$

$dw /= m.$

Andrew Ng