LinearAlgebra

Tuesday, July 16, 2019 9:06 AM

Some quick review from yesterday's notes

$\bar{X} = (x_0, \dots, x_{N-1}) \quad x \in \mathbb{R}$

$\|\vec{x}\| = \sqrt{\sum_{0}^{N-1} x_i^2}$

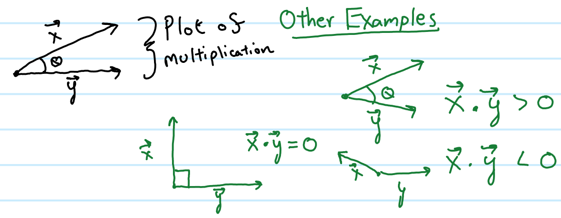
Assume x is the list described on the left, the magnitude of x is described as:

Another important geometric concept to know about is dot products:

Formula:

$\vec{x} \cdot \vec{y} = \underbrace{\|\vec{x}\| \|\vec{y}\|}_{\text{magnitude of x times magnitude of y}} \underbrace{\cos\theta}_{\text{multiplied with cosine of angle between them}}$

$\Longrightarrow$



Try some examples of dot products in a blank notebook now. Just as a useful hint, when doing this in the future, make use of np.matmul(x,y) to get the dot product or np.linalg,norm(x) to get the normalized value of x to find the dot product in a little lengthier process.

Matrix Vector Multiplication

$$3\left\{\overbrace{\begin{bmatrix} x_0 & y_0 \\ x_1 & y_1 \\ x_2 & y_2 \end{bmatrix}}^{2}\right. \qquad \overset{(2,)}{[\bar{x}, \bar{y}]} \quad \begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$$

3x2 (reshape axis)

We multiply all the green rows with the blue column

Summary of steps of matrix multiplication

- Assume you have 2 matrices in the forms (nxm) and (k,i) respectively
- For the matrices to be able to be multiplied, m & k have to be equal in value
- The size of the final vector will be (nxi)

Review of Loss Function Process from Yesterday

Form of Function

$f(\underbrace{\{\omega\}}_{\text{weights aka. learnable parameter}}; \underset{\text{data}}{x}) \longrightarrow y_{prediction}$ we then go towards $\xrightarrow{\text{loss function}}$ $\mathcal{L}(y_{prediction}, y_{true})$

we then use this to update our parameters

$\omega \leftarrow \omega^* - \delta \left.\frac{d\mathcal{L}}{d\omega}\right|_{\omega^*}$ $\longrightarrow$ δ always > 0 (though keep step sizes low)

This δ is the learning rate.

This is a hyper-parameter, which means it's a value that we set

Universal Function Approximator

Suppose

$f(\vec{x}) \longrightarrow y$, where $\vec{x} \in \mathbb{R}$ $\vec{x} = (x_0, \dots, x_{N-1})$ $y \in \mathbb{R}$, we want to find $\underbrace{F \approx f}_{\text{the function that represents } f}$

The universal function approximator theorem states that:

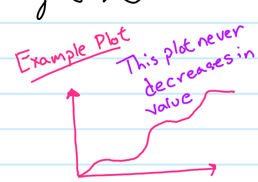
There exists integer N and $\{\vec{\omega}_i\}, \{v_i\}, \{b_i\}$ for any $|F(\{\vec{\omega}_i\}_{i=0}^{N-1}, \{b_i\}_{i=0}^{N-1}, \{v_i\}_{i=0}^{N-1}; x) - f(x)| < \epsilon$ for any $\epsilon > 0$

The function states that:

$$F(\{\vec{\omega}_i\}_{i=0}^{N-1}, \{b_i\}_{i=0}^{N-1}, \{v_i\}_{i=0}^{N-1}; x) = \sum_{i=0}^{N-1} v_i \varphi(\vec{x} \cdot \vec{\omega}_i + b_i)$$

where $\varphi(x) = \frac{1}{1+e^{-x}}$

Where $\varphi(Q)$ is any continuous, monotonically-increasing function that is bounded



$F(\{\vec{w}_i\}_{i=0}^{N-1}, \{b_i\}_{i=0}^{N-1}, \{v_i\}_{i=0}^{N-1}; x) = \sum_{i=0}^{N-1} v_i \varphi(\vec{x}\cdot\vec{w}_i + b_i)$

monotonically-increasing
function that is bounded

<u>Now work through the Universal Function Approximator Notebook</u>