Deep Learning & Pytorch Basics

Zhaoxuan Tan
LUD lab
Xi’an Jiaotong University

Janunary 19, 2022

Table of Contents

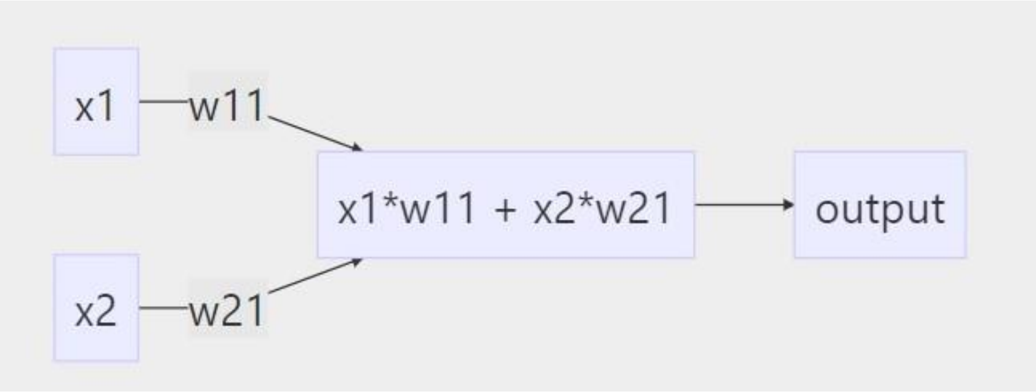
- Linear Layer / Multilayer Perceptron (MLP)

- Pytorch Basics

Feature Vector

- Bot detection for example
  - Follower count
  - Friend count
  - Followee count
  - Tweet count
  - Verified
  - …

Matrix Multiplication



$$\begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ b_0 & b_1 & b_2 & b_3 \\ c_0 & c_1 & c_2 & c_3 \\ d_0 & d_1 & d_2 & d_3 \end{bmatrix} . \begin{bmatrix} w_{00} & w_{01} \\ w_{10} & w_{11} \\ w_{20} & w_{21} \\ w_{30} & w_{31} \end{bmatrix}$$

Linear Layer

- Weight
- Bias
- input / output dimension

$$y = Wx + b$$

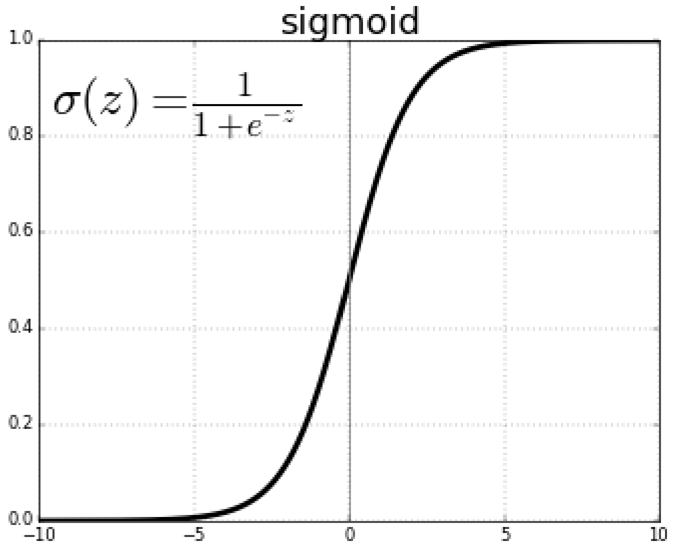
Activation Function

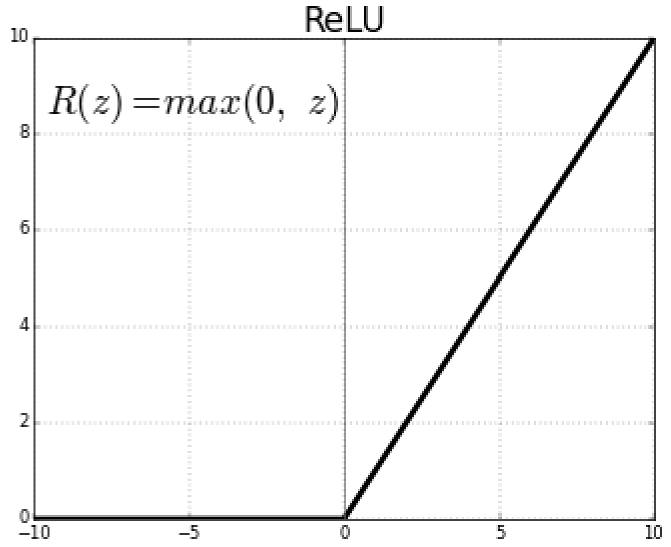
- ReLU (Rectified Linear Unit)

$$f(x) = \max(0, x)$$

- Sigmoid

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





- Tanh?
- Leaky ReLU?

Forward Pass of 2-layer NN

Classification with NN

- Softmax function

$$g(y)_j = \frac{e^{y_j}}{\sum_{k=1}^{K} e^{y_k}}$$

Loss Function

- Cross Entropy Loss

$$L_{CE} = \frac{1}{N}\sum_{i} L_i = -\frac{1}{N}\sum_{i}\sum_{c=1}^{M} y_{ic}\log(y_{ic})$$

- Binary Cross Entropy Loss

$$L_{BCE} = \frac{1}{N}\sum_{i} L_i = -\frac{1}{N}\sum_{i}\sum_{c=1}^{M}[y_i\log(y_i)+(1-y_i)\log(1-y_i)]$$

Gradient Descent

$$L(\theta+\varepsilon) \approx L(\theta) + \varepsilon L'(\theta)$$

$$L(\theta-\eta L'(\theta)) \approx L(\theta) - \eta L'(\theta)^2$$

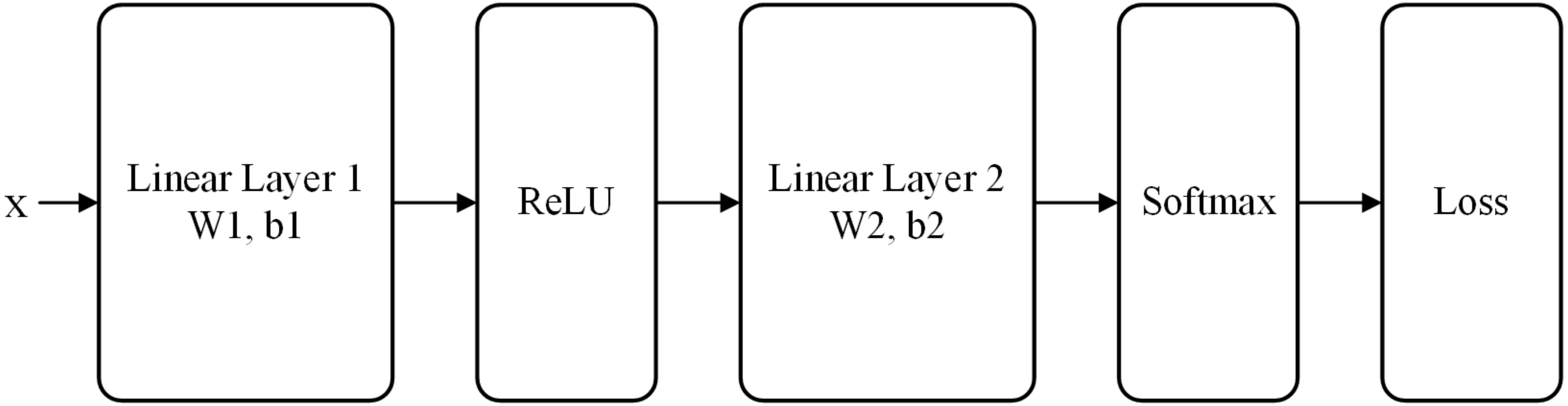
$$L(\theta-\eta L'(\theta)) \leq L(\theta)$$

$$\theta \leftarrow \theta - \eta L'(\theta)$$

- Learning rate

Backward Pass

- Chain rule $$\frac{dy}{dx} = \frac{dy}{du} \cdot \frac{du}{dx}$$



Optimizer

- Gradient Descent

$$\nabla L(\theta) = \frac{1}{M}\sum_{i=1}^{M} \nabla L(f(x_i, \theta), y_i)$$

- Stochastic Gradient Descent (SGD)

$$\nabla L(\theta) = \frac{1}{m}\sum_{j=1}^{m} \nabla L(f(x_j, \theta), y_j)$$

- Adam?

Overall: 2-layer MLP

```
for epoch in epochs:
    for batch in batches:
        get input feature vectors in batch
        forward pass
        loss function
        backward pass
        gradient descent
    end batch
end epoch
```

Pytorch Implementation

- Hyperparameter tunning
  - train / valid / test set

Thx for Attention

Zhaoxuan Tan

LUD lab

Xi’an Jiaotong University

Janunary 19, 2022