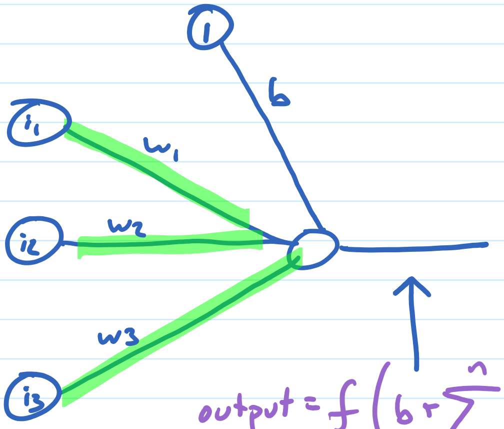
ANN Parameters

training data — used to adjust weights during training

validation data — used to test as you adjust NN parameters

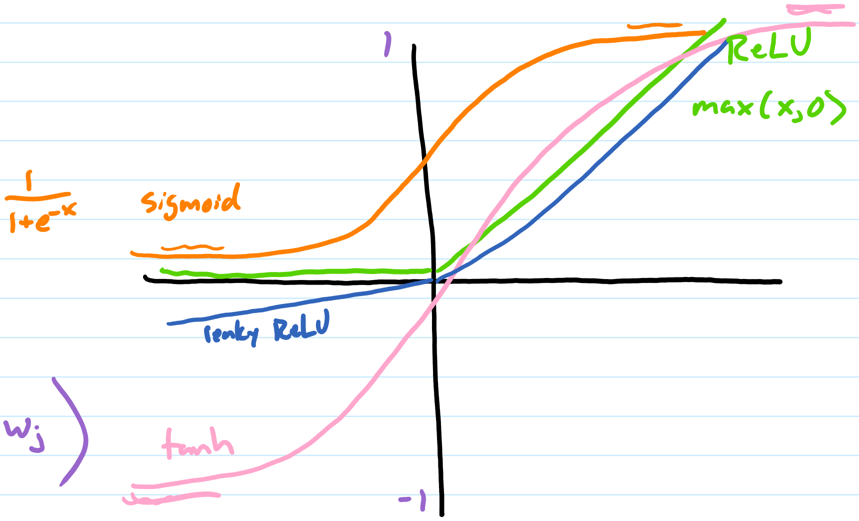
test data — used to evaluate NN after training (on data it hasn't seen before)

Activation Function



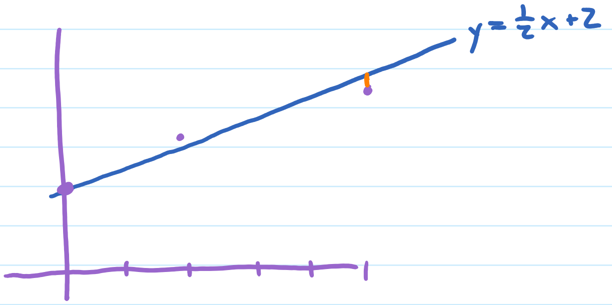
$$\text{output} = f\left(b + \sum_{j=1}^{n} i_j \cdot w_j\right)$$

activation function



Overfitting

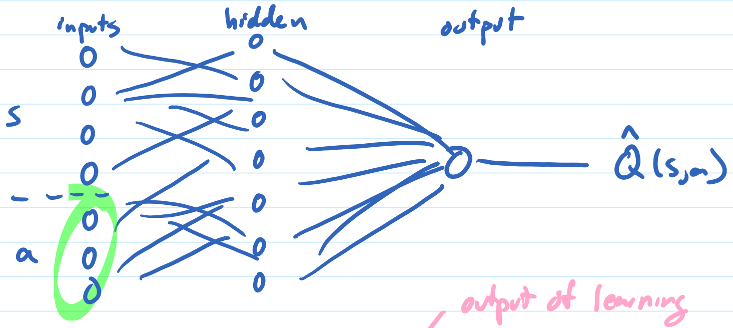
input	output	
0	2	
1		
2	3.1	3.0
5	4.3	4.5



```
~~if x == 0~~
    ~~return 2~~
~~else if x == 2~~
    ~~return 3.1~~
~~else~~
    ~~return 4.3~~
```

Deep Q Learning

> 1 hidden layer



$$Q(s,a) = \sum_{s \to s'} P(s,a,s') \cdot (R(s,a,s') + V(s'))$$

$\max_{a'} Q(s',a')$

output of learning

$\hat{Q}((180,4,10,24),1) = .38$

$\hat{Q}((180,4,10,24),2) = .54$

$\hat{Q}((180,4,10,24),3) = .21$

$\hat{Q}((180,4,10,24),4) = .04$

reward 0
new state (64,4,10,22)

$r + \max_{a'} \hat{Q}(s',a')$

$0 + 0.68$

output of target

adjust weights on this

```
initialize learning, target networks
    used to set values you train the learning network towards
for each iteration
    for each of n episodes        smallish value
        for each event
            add (s,a,s',r) to replay database

    sample replay database

    train learning network toward $r + \max_{a'} \hat{Q}_{target}(s',a')$ over chosen samples

    if enough time passed
        copy learning network to target network
```