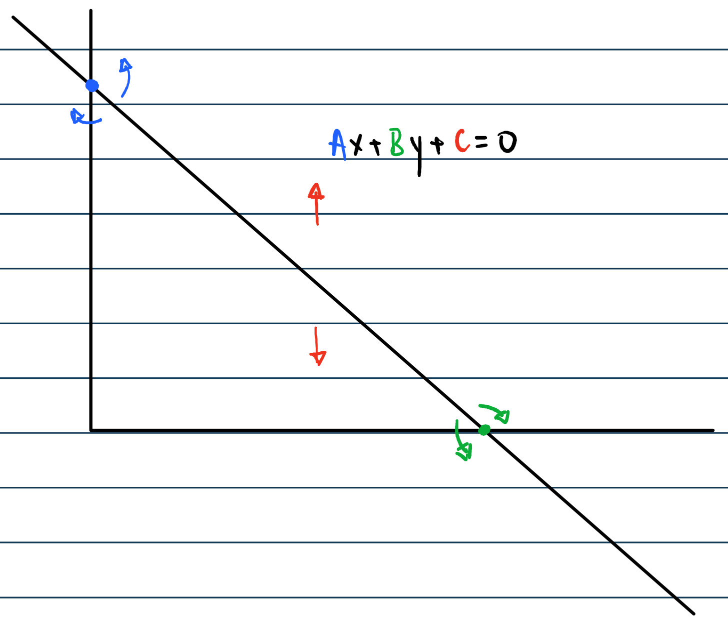
Perceptron Trick

Tuesday 30 April 2024 4:04 PM

→ Won't find the best possible solution

1) Start with random values of A, B & C

2) run loop → 1000
   - check how many points missclassified [optional]
   - Choose a random data point
   - Ask if it has been classified correctly
     - Yes — line unchanged
     - NO — transform line so that data point is on the correct side of the line.



→ To move line towards positive region if a negative point is in positive region

$$\begin{array}{ccc} A & B & C \\ -\eta(x_i & y_i & 1) \\ \hline A_{new} & B_{new} & C_{new} \end{array}$$

→ To move line towards negative region if a positive point is in negative region.

$$\begin{array}{ccc} A & B & C \\ +\eta(x_i & y_i & 1) \\ \hline A_{new} & B_{new} & C_{new} \end{array}$$

Algorithm:

```
epoch → , η →

    W = [W_0 W_1 ... W_n]
    for i in range(epochs):

        select a point randomly

            if $y_{point} \in$ -ve region & $\sum_{i=0}^{n} W_i x_i \geq 0$  1
```

→ step function

if $y_{point} \in -ve$ region & $\sum_{i=0} w_i x_i \geq 0$ 1

$$W_{new} = W - \eta\,[x_0\ x_1 \ldots x_n]$$

if $y_{point} \in +ve$ region & $\sum_{i=0}^{n} w_i x_i < 0$ 1

$$W_{new} = W + \eta\,[x_0\ x_1 \ldots x_n]$$

$$\Rightarrow W_{new} = W + \eta\,(y_i - \hat{y}_i)\,x_i$$

→ step function

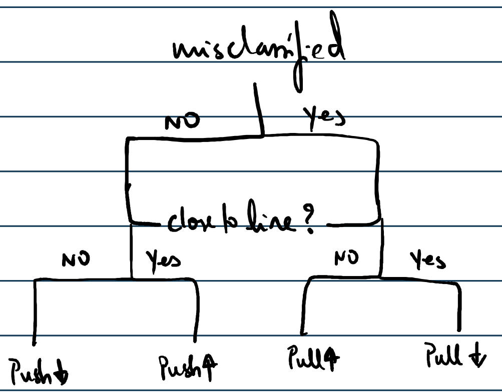
from sklearn.datasets import make_classification

→ overfitting

→ A better approach?

→ point misclassified? YES PULL / NO PUSH

How much to pull or push?



sigmoid function

$$W_{new} = W + \eta\,(y_i - \hat{y}_i)\,x_i$$

$$\hat{y}_i = \sigma(z)$$

where $z = \sum w_i x_i$

→ Classes play a push & pull game

→ Can we do better?