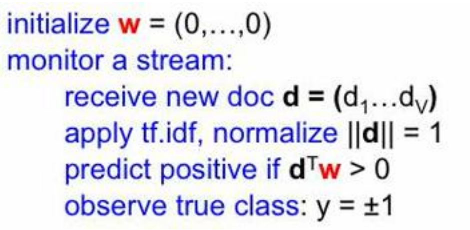
What is PassiveAggressiveClassifier ?

Passive Aggressive algorithms are online learning algorithms. Such an algorithm remains passive for a correct classification outcome, and turns aggressive in the event of a miscalculation, updating and adjusting. Unlike most other algorithms, it does not converge. Its purpose is to make updates that correct the loss, causing very little change in the norm of the weight vector.

How it works ?

Initialize the weight vector to zeros, so we have zero for every term.

Looking at the stream of data we get a new document, it’s a vector over the vocabulary (that’s why we use Tfidf), then to make it prediction we multiply by the weight vector and see if it’s positive or negative, (at the beginning it’s zero because we haven’t learning yet) then we observe the true class of the document: y=+/- 1 (positive or negative).

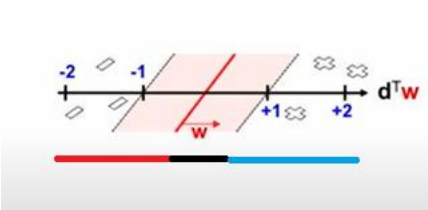


```
initialize $\mathbf{w} = (0,\ldots,0)$
monitor a stream:
    receive new doc $\mathbf{d} = (d_1 \ldots d_V)$
    apply tf.idf, normalize $||\mathbf{d}|| = 1$
    predict positive if $\mathbf{d}^T\mathbf{w} > 0$
    observe true class: $y = \pm 1$
```

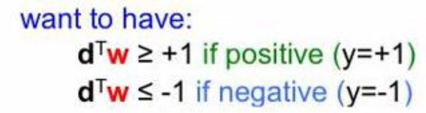
Suppose that the document is positive (Y=+1), if the document scores on the left side of the decision boundary (1), so the prediction is wrong and we want to penalize that decision and change the wave filter.

If the document falls too close to the buffer zone (2), we want to penalize it too, even if it was classified correctly.

If the document falls into that zone (3), it’s fine, and we don’t need to change W.



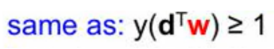
How can we encode that ?



want to have:

$$\mathbf{d}^T\mathbf{w} \geq +1 \text{ if positive } (y=+1)$$

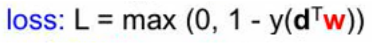
$$\mathbf{d}^T\mathbf{w} \leq -1 \text{ if negative } (y=-1)$$



same as: $y(\mathbf{d}^T\mathbf{w}) \geq 1$

-> That’s what we want to enforce

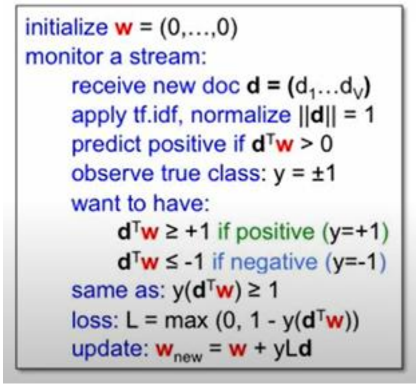
So we could compute a loss function:



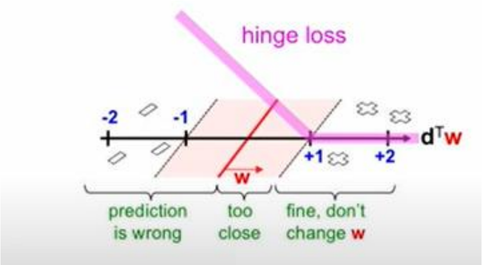
loss: $L = \max(0, 1 - y(\mathbf{d}^T\mathbf{w}))$

Basically we would like that product d transpose W times plus or minus 1 to be over 1 always, if we have something that it's smaller than 1 we are going to take that as a loss, if we have something that is 1 or bigger than 1 we ignore it, that what max represents.

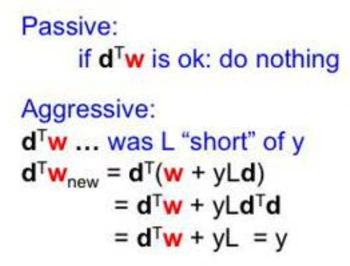
So L is the penalty.



```
initialize $\mathbf{w} = (0,\dots,0)$
monitor a stream:
    receive new doc $\mathbf{d} = (d_1 \dots d_V)$
    apply tf.idf, normalize $||\mathbf{d}|| = 1$
    predict positive if $\mathbf{d}^T\mathbf{w} > 0$
    observe true class: $y = \pm 1$
    want to have:
        $\mathbf{d}^T\mathbf{w} \geq +1$ if positive $(y=+1)$
        $\mathbf{d}^T\mathbf{w} \leq -1$ if negative $(y=-1)$
    same as: $y(\mathbf{d}^T\mathbf{w}) \geq 1$
    loss: $L = \max(0, 1 - y(\mathbf{d}^T\mathbf{w}))$
    update: $\mathbf{w}_{new} = \mathbf{w} + yL\mathbf{d}$
```



The way PA is using the loss as effectively the learning rate.



Passive:

if $\mathbf{d}^T\mathbf{w}$ is ok: do nothing

Aggressive:

$\mathbf{d}^T\mathbf{w}$ … was L "short" of y

$$\begin{aligned}\mathbf{d}^T\mathbf{w}_{new} &= \mathbf{d}^T(\mathbf{w} + yL\mathbf{d}) \\ &= \mathbf{d}^T\mathbf{w} + yL\mathbf{d}^T\mathbf{d} \\ &= \mathbf{d}^T\mathbf{w} + yL \ = y\end{aligned}$$