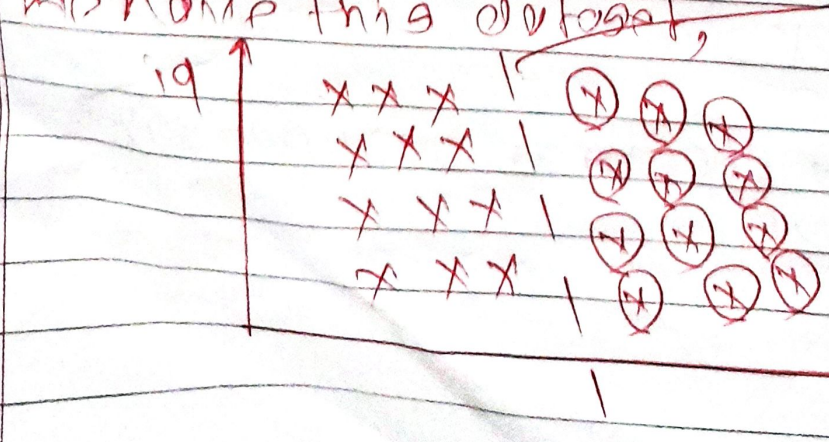
Sckit learn's implimentation.

```
from skearn.linear_model import Perceptron

P= Perceptron()
P.fit( x, y)  # put both x and y value init

P.coef_  # (w1, w2) value
P.intercept_  # (b) value

# To plot decision regions seporated by
  perceptron line boundry.

from ml.extend.plotting import plot_
decision_regions

plot_decision_regions (X.values, Y.values,
                       clf = P, legends = 2)
```

perceptron trick

- This is jugadu technique to find weight and biases.
- We have this dataset,



We want to find the line equation basically

First observation is this data is linearly seperable.

- This line equation is not in this formate, $y = mx + c$ but this is in this formate that is general formate $AX + BY + C = 0$ (which is used in logistic regression)

$X_1 \Rightarrow$ cgpa
$X_2 \Rightarrow$ IQ
$C \Rightarrow$ y-intercept

So, this line equation becomes,

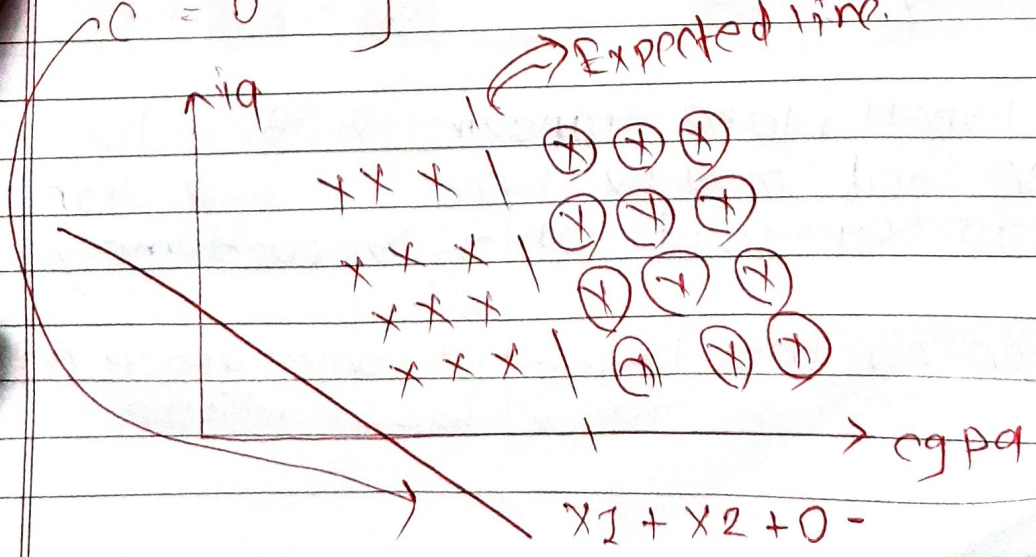
$$AX_1 + BX_2 + C = 0$$

- our end objective is we have to find this 'A', 'B' and 'C' value to classify those point by drawing a line.

- <u>solution is very simple,</u> we started from random value of A, B and C.

$A = 1$
$B = 1$
$C = 0$

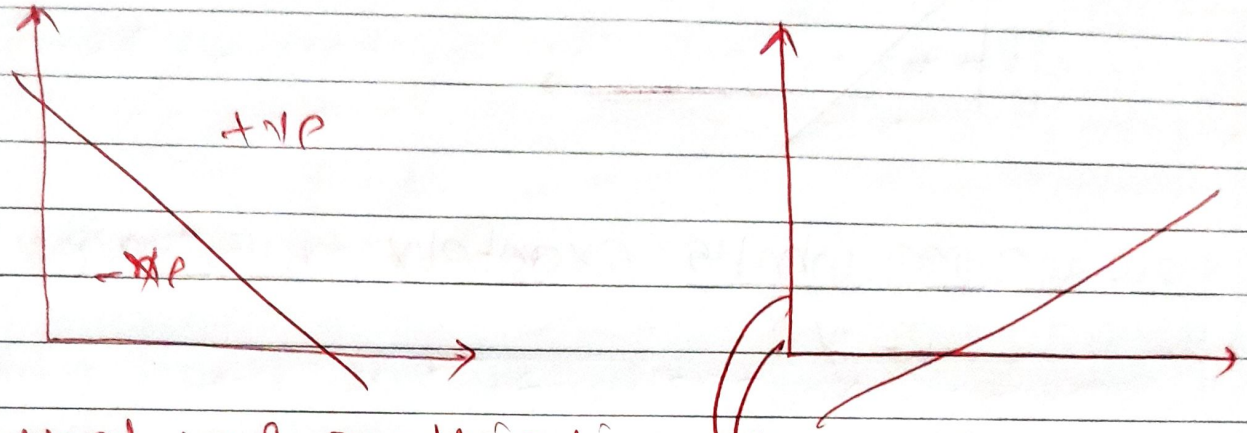
} that is we start from a random line



- In summary, we ask random point this line according to you is right or wrong. modified line according to this point (wrong classified by our line)
- We have to move this line, (clearly this line not good classifier) By using loop
- Generally we run a loop 1000 time we can decrease or increase it as we want.
- Every loop we select a random student/point and ask this point according to you this line was right or wrong.
  $(X1 + X2 + 0) \Rightarrow$ (X) this point
  - if this point satisfied by this line we donot do anything $\Rightarrow$ (Y)
  - Now we ask (X) 'X' this point, it tells we have problem
- After this we make such transformation in line $(X1 + X2 + 0)$, 'X' this point classified correctely.
- So againe we ask random point and modified for wrongely classified point, repetedely.
- In nutshell, take random A, B and C value for line and modify line in loop for misclassified point which is randomely choose
- We can run this loop until convergence happens Or we can decides loop value.

How this transformation happens of line?

How to identify positive region and negative region?

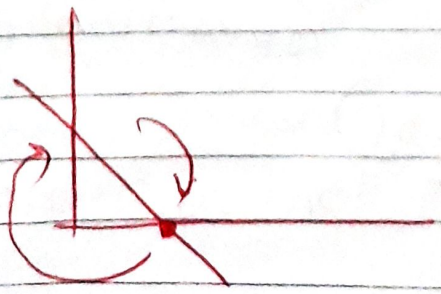


- What is f for this line.

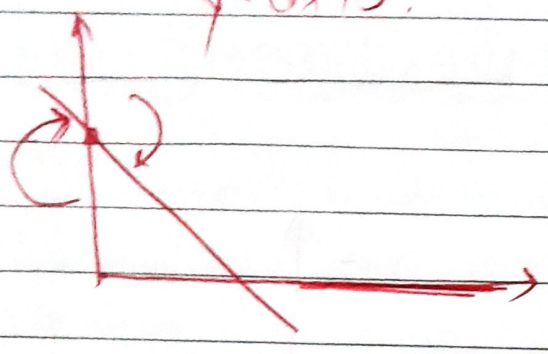
$Ax+By+C>0$ +ve region
$Ax+By+C<0$ -ve region

- See in Desmos.

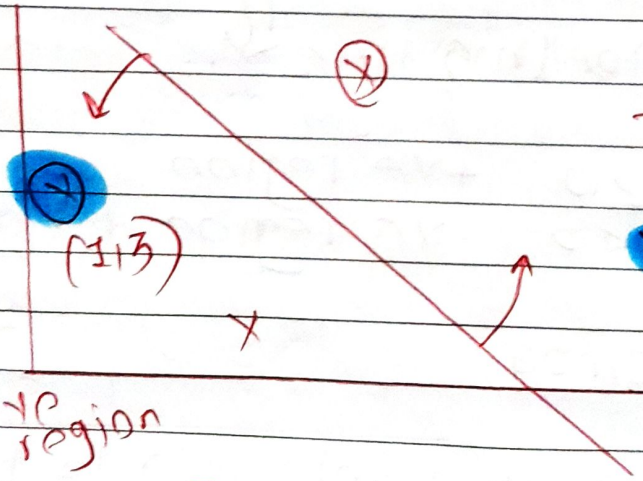
- Now back to our point, how to transform a line, Line में change mainly आता है by changing A B and C value.
  - C increase or decrease line parallely up's and down no change in slope.
  - B increase or decrease line rotate's around x-axis



- A increase or decrease, rotation around y-axis.



※ Now we are what's exactly transformation going to apply!



- This are (X) and (X) miss classified है, to transformation हम इन दोनो के case में करना होगा.
- First for this point (X) (4,5), add one in this co-ordinate (4,5,1) ⇒ positive point ~~negative~~ region

$$\begin{array}{ccc} 2 & 3 & 5 \\ -4 & -5 & -1 \\ \hline -2 & -2 & 4 \end{array} \quad (A,B,C)$$

(-) because we want to move line in -ve region

- Now new ~~coordinate~~ for our line is,

$-2x - 2y + 4 = 0$

- After this, our point (4, 5) goes to negative region.
  ⇒ negative point in +ve region
- Similar for this point (1, 3) ⊕ is because we want to move in (+ve) direction

$$\begin{array}{r} 2\ 3\ 5 \\ +\ 1\ 3\ 1 \\ \hline 3\ 6\ 6 \end{array}$$

- We don't do this much of transformation, in machine learning, instead we multiply co-ordinates of point by learning rate,

learning rate = 0.01

$$\begin{array}{r} 2\ \ 3\ \ 5 \\ 0.01\times1\ \ 0.01\times3\ \ 0.01\times1 \\ \hline 2.01\ \ 3.03\ \ 5.05 \end{array}$$

New coefficient = old − η co-ordinates

Algorithm

(X0) cgpa	(X1) iq	(Y) placed
7.5	81	1
8.9	109	1
7.0	81	0

$W0 + W1X1 + W2X2 = 0$

$W0 = b$

$W1 = A$

$W2 = B$

- We do a here little simplification, add x_0.

x_0	x_1 (cgpa)	x_2 (iq)	y (placed)
1	7.5	81	1
1	8.9	109	1
1	7.0	81	0

- $(w_0 + w_1x_1 + w_2x_2 = 0)$ => Previous equation
- $(w_0x_0 + w_1x_1 + w_2x_2 = 0)$ => After simplification.
- We now simplify above into,

$$\sum_{i=0}^{2} w_i x_i = 0$$

This is general equation of our line/model.

- We have w_0, w_1 and w_2 value here,
- Now we put this value in this equation to predict (y placed / not)

$$w_0x_0 + w_1x_1 + w_2x_2 = 0$$

This will be 'z' value.

- if $z \geq 0$ then 1 (y) and if $z < 0$ then 0 (y).

- We can represent this as a matrix, $[w_0\ w_1\ w_2]$ and $\begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$ this intern ally dot product of

$$\begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix} [w_0\ w_1\ w_2] . \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$$

If this multiplication is gretter than 0 then Y is 1 and vice versa,

- Basically हम epoch value set करेंगे (1000), $\eta = 0.01$ बादमे हम loop चलायेंगे,

This algorithm pseudo code.

```
For i in range (epochs):
    In this loop we select a point
    which are randomely selected,
    if ( (xi) /student/ row i selected
        belongs to negative region
        and Σ (i=0) wi xi ≥ 0
    ((
        according to model
        this is positive.
    in if we update this,
    wnew = wold - η [xi | x0 x1 x2]
    if ( xi ∈ P and Σ (j=1) wj xj < 0)
```

$$w_{new} = w_{old} + \eta x_i$$

Simplified Algorithm

- $x_i \in N$ and $\sum w_i x_i \geq 0$

$$w_n = w_0 - \eta x_i \quad \text{---(1)}$$

$x_i \in P$ and $\sum w_i x_i < 0$

$$w_n = w_0 + \eta x_i \quad \text{---(2)}$$

- Instead of applying this condition we apply only one condition

for in 1000
random student

$$w_n = w_0 + \eta (y_i - \hat{y}_i) x_i$$

- what y_i and $\hat{y}_i$

y_i (actual)	$\hat{y}_i$ (predicted by model)	$y_i - \hat{y}_i$
1	1	0
0	0	0
1	0	1
0	1	-1

This four combination only exist

- Let's see first case, $y_i = 1$ and $\hat{y}_i = 1$, $(y_i - \hat{y}_i) = 0$

$$w_n = w_o$$

- second $y_i = 0$ and $\hat{y} = 0$ same, like above

- third case, $y_i = 1$ and $\hat{y}_i = 0$, this is positive student in negative region, now

$$w_n = w_o + \eta (y_i - \hat{y}_i) x_i$$

$$w_n = w_o + \eta x_i \qquad (y_i - \hat{y}_1) = 1$$

... (this is similar to second equation) equals to equation (2)

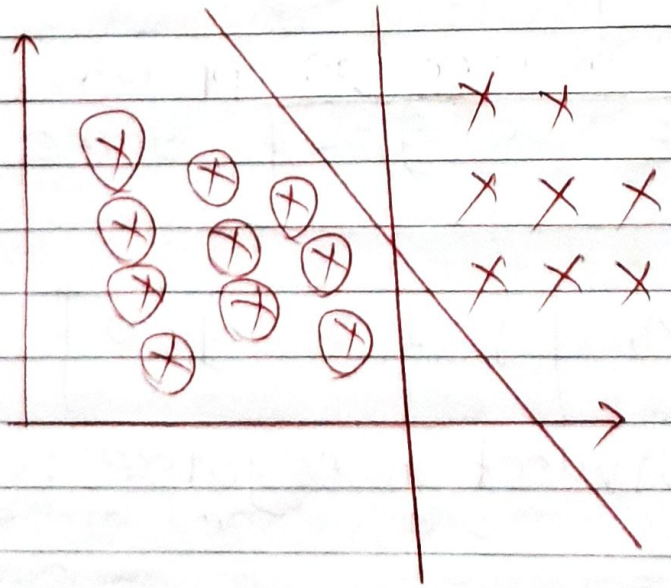
- fourth case, $y_i = 0$ and $\hat{y}_i = 1$, now,

$$w_n = w_o - \eta x_i$$

... (this is similar to first equation) equals to equation (1).

Perceptron trick Problem

1. We cannot sure that w_0, w_1 and w_2 values are right values to separate this ⊗ and × this point.



- We could get one of two lines by perceptron trick but we cannot sure which line was best.
- That's why we cannot quantify the result.

2. When we find this value w_0, w_1 and b we use loss function, because of loss function we can able to quantify the result.