GENERATIVE ADVERSARIAL NETWORK (GAN)

Adversarial → conflict or opposition

GAN are deep neural net architectures comprised of two neural networks, competing one against the other (i.e why adversarial)

→ GAN are neural networks that are trained in an adverserial manner to generate data mimicking same distribution.

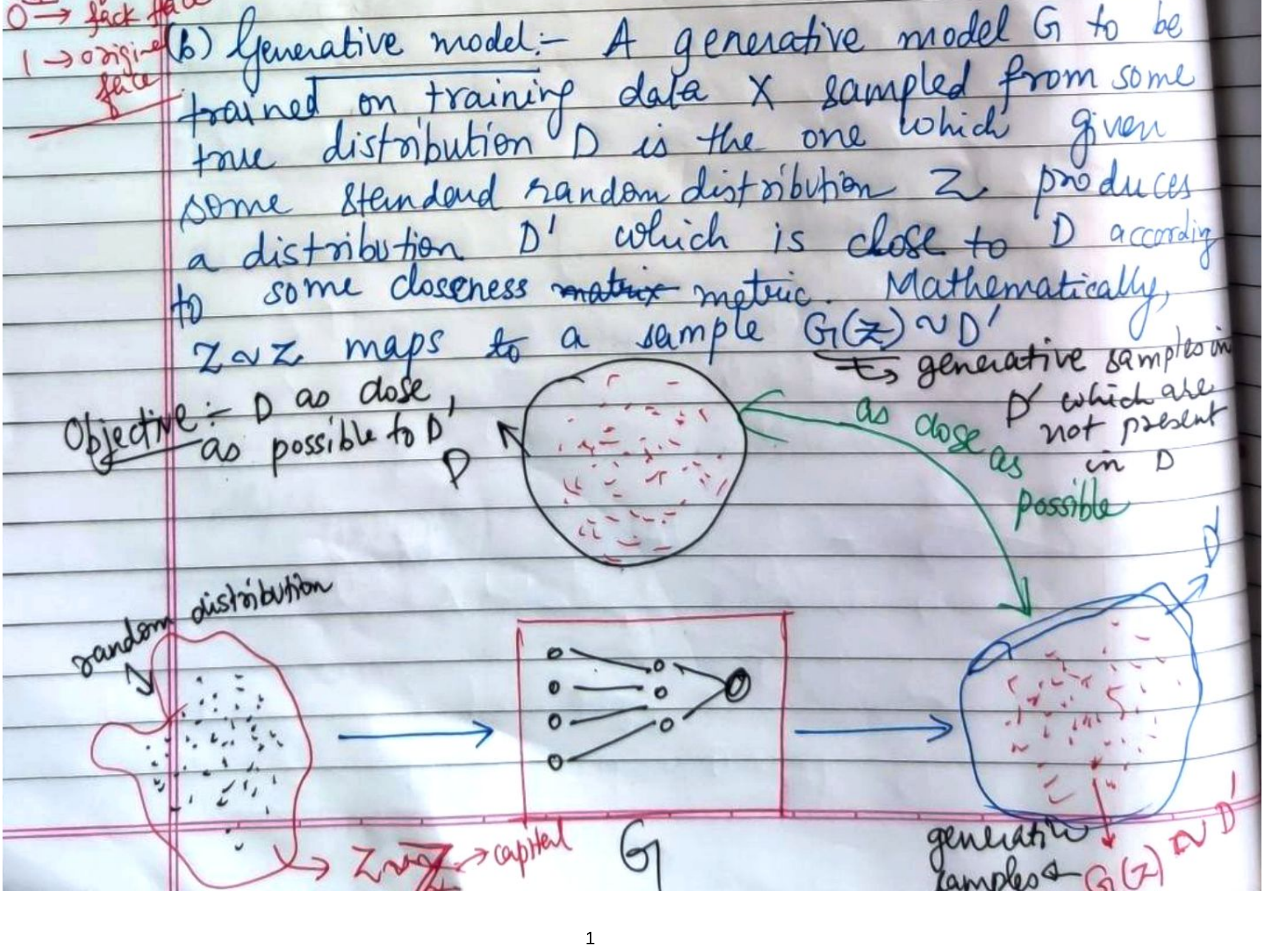
→ Two classes ~~mod~~ of models in machine learning

To discriminate between a fack face & original face
output
0 → fack face
1 → original face

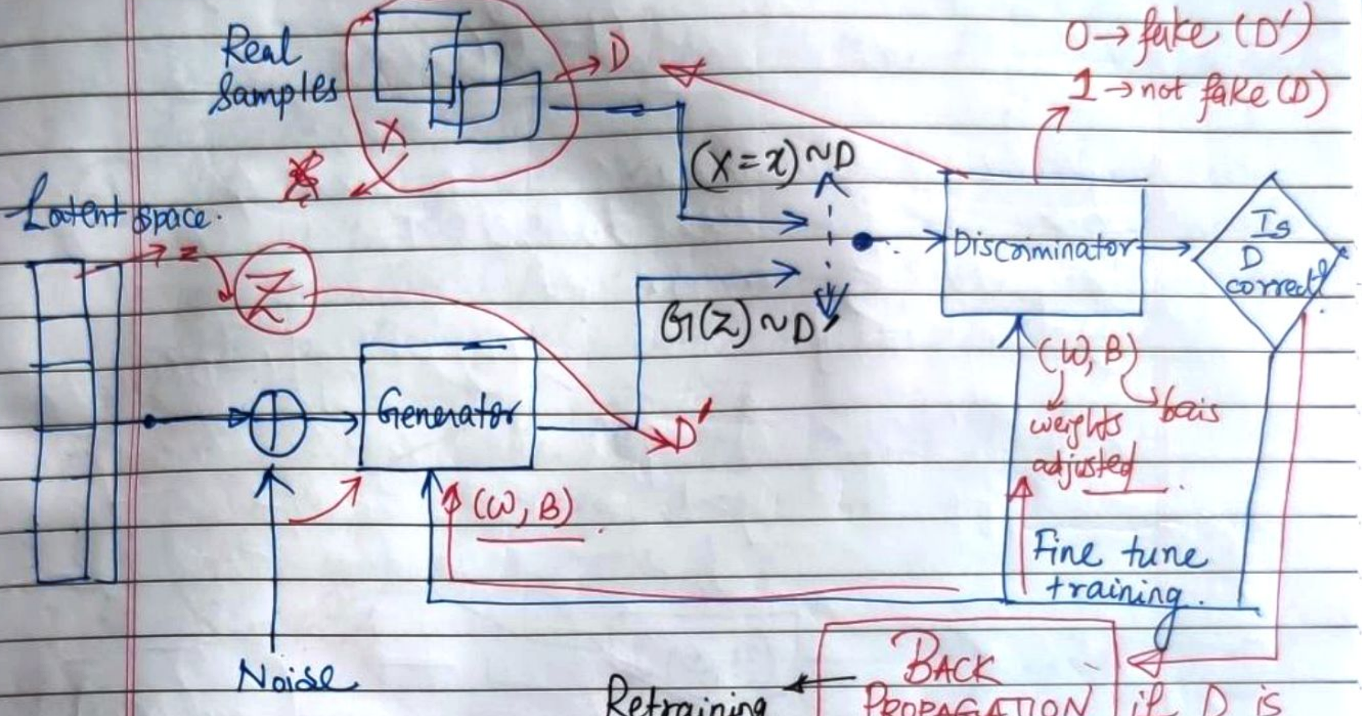
(a) Discriminative model:— It is the one that discriminates between 2 different classes of date

(b) Generative model:— A generative model G to be trained on training data X sampled from some true distribution D is the one which given some standard random distribution Z produces a distribution D' which is close to D according to some closeness ~~matrix~~ metric. Mathematically, $z \sim Z$ maps to a sample $G(z) \sim D'$ → generative samples in D' which are not present in D

Objective:- D as close as possible to D'



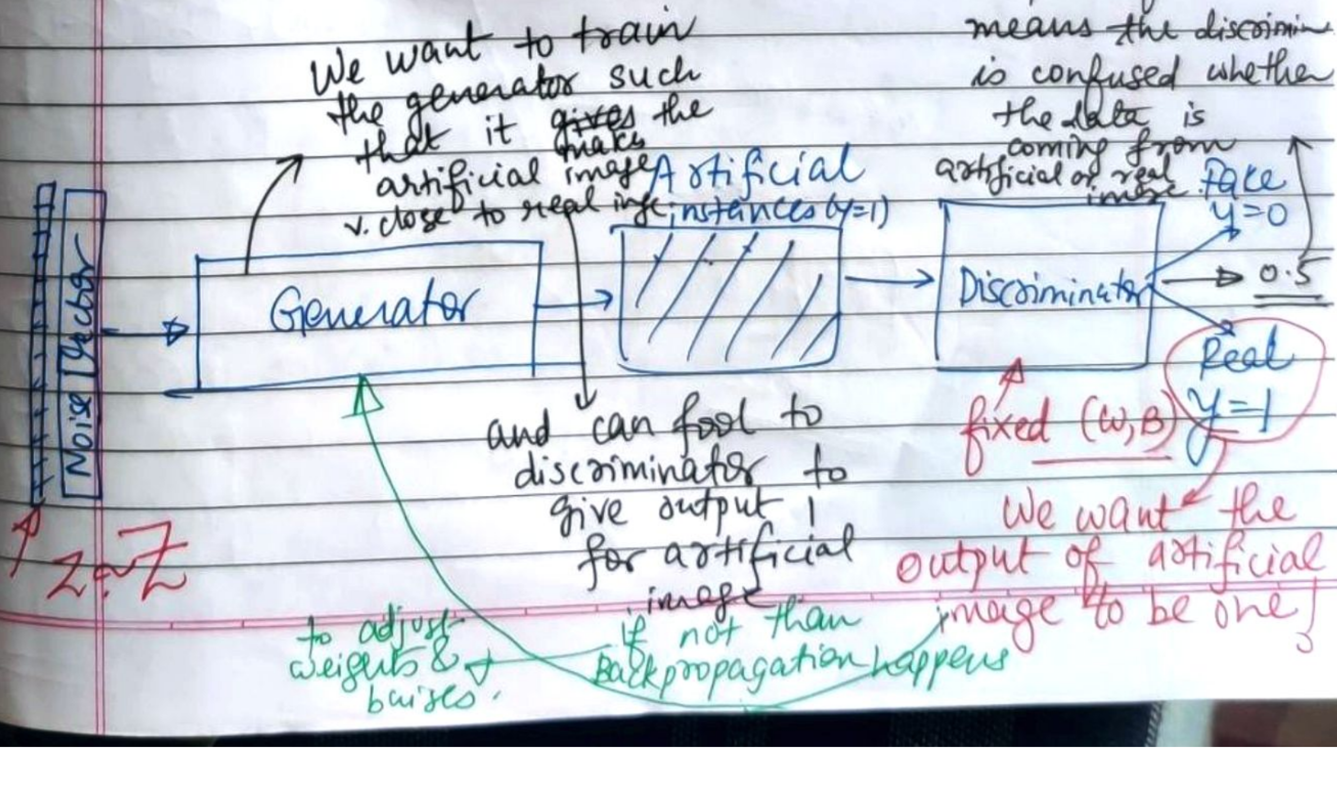
GENERATIVE ADVERSERIAL NETWORK BLOCK DIAGRAM



Retraining

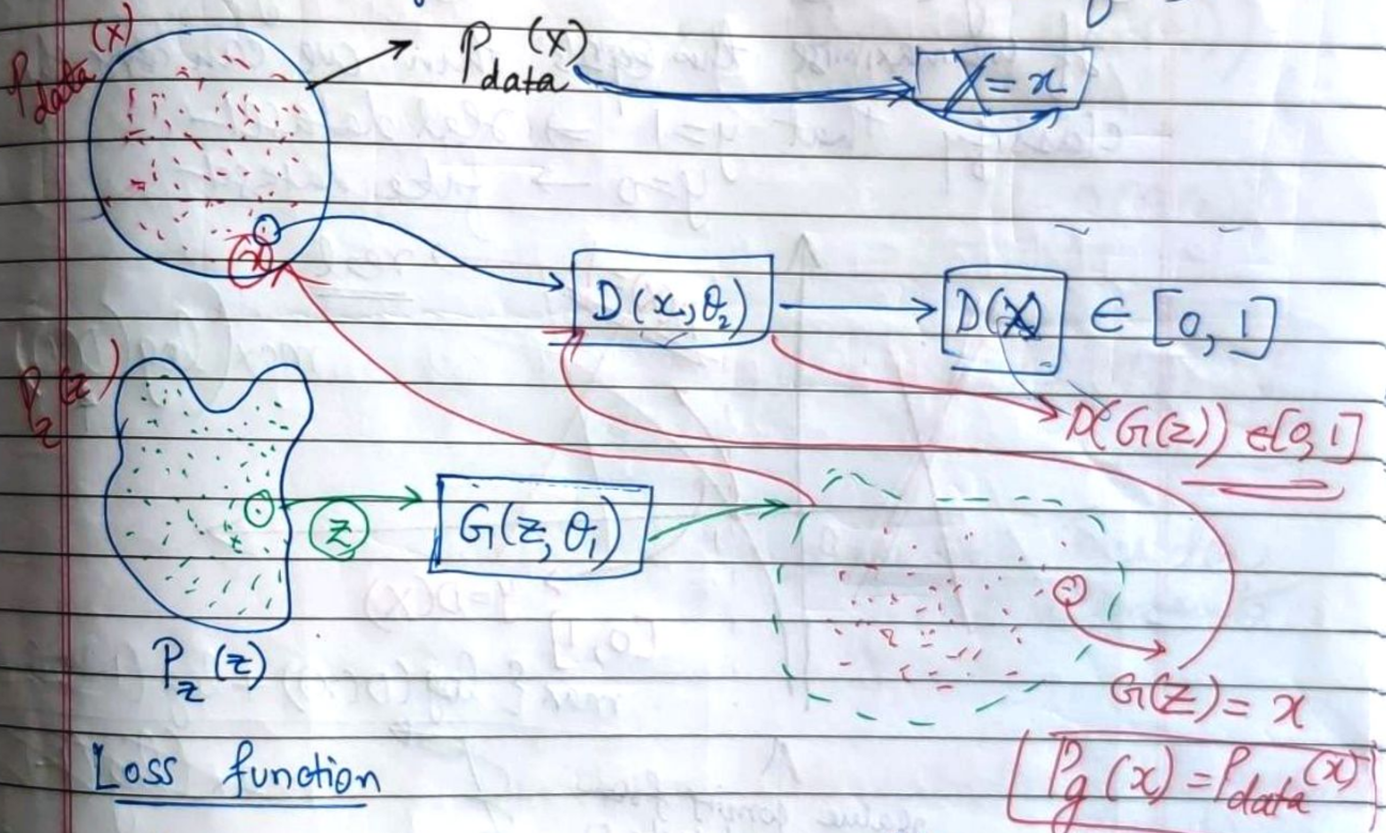
to make fake as Real sample indistinguisheble! so $D' \sim D$ so that it will output 0.5 which is neither 0 or 1 → neither fake or real.

Training the Generator



Loss Function Of GAN

1. **Discriminator** :- Role is to distinguish between actual data & fake data
2. **Generator** :- Role is to create data in such a way so that it can fool the ~~generato~~ discriminator (so that discriminator will not be able to distinguish between Real & fake)



Loss function

$$L(\hat{y}, y) = [y \log \hat{y} + (1-y) \log(1-\hat{y})]$$

$\hat{y}$ = reconstructed image

y = original image.

The label for the data coming from $P_{data}(x)$ is $y=1$ (real)

& $\hat{y} = D(x)$ so putting this we obtain

$$L(D(x), 1) = \log(D(x)) \longrightarrow (A)$$

& for data coming from generator, The label is $y=0$ (artificial) i.e.

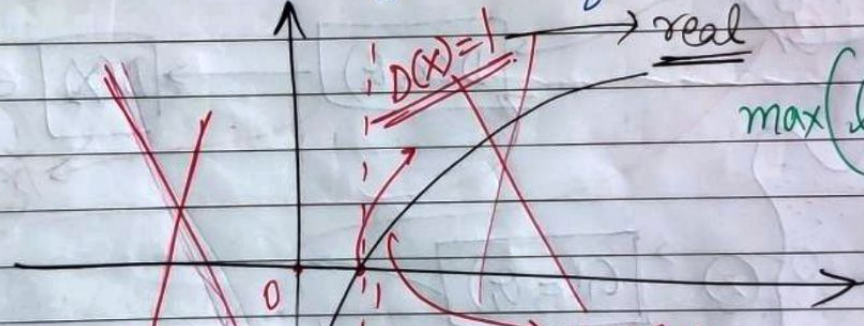
$\hat{y} = D(G(z))$ so in that case

$$L(D(G(x)),0) = (1-0)\log(1-D(G(z)))$$
$$= \log(1-D(G(z))) \quad \text{—— B}$$

Objective of discriminator is to correctly classify fake v/s the real dataset. For this A & B should be maximised

A $[\log(D(x))]$ B $[\log[1-D(G(z))]]$

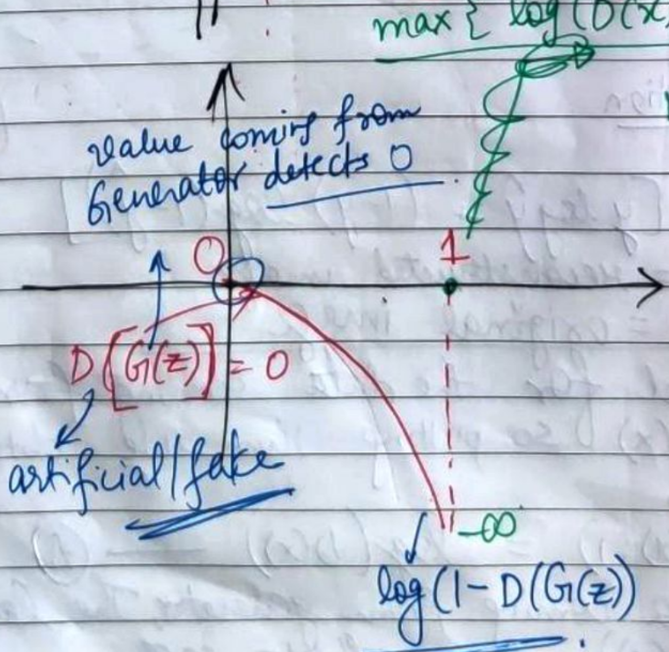
If we maximise this eqn's than we can correctly classify that $y=1 \rightarrow$ real dataset
$y=0 \rightarrow$ fake dataset



$\max(\log(D(x)))$

$\max\{\log(D(x)) + \log(1-D(G(z)))\}$

~~$\max\{\log(1-D(G(z)))$~~



The target of generator is to prove the probability of $D(G(z)) \simeq 1$ to fool the discriminator and make artificial as real image

$y = 0 \Rightarrow$ fake $\hat{y} = D(G(z))$ To get $D(G(z)) = 1$.

$L(D(G(z)), 0) = \min \{ (1) \log [1 - D(G(z))] \}$ minimise

For the target of generator to fool discriminator it happens when $\log(1 - D(G(z)))$ tends to $-\infty$.

So, we minimise that fn.

$$\boxed{\min [\log(D(x)] + \log(1 - \underbrace{D(G(z))}_{1}]}$$

NOTE:

1) Task of discriminator is to maximise

$$\max [\log(D(x)) + \log(1 - D(G(z)))$$

2) Task of generator is to minimise

$$\min [\log(D(x)) + \log(1 - D(G(z)))$$

3) Combining both

$$\boxed{\min_{G} \max_{D} [\log[(D(x))] + \log[1 - D(G(z))]]}$$

Considering all the elements, here we consider only one element of all, for all elements we get the following expression

$$\boxed{\min_{G} \max_{D} V(D, G) = \{ E_{x \sim p_{data}(x)} [\log(D(x))] + E_{z \sim p_z(z)} [\log(1 - D(G(z)))]}$$