

# GAN semi-supervised : SGAN

## Architecture-fonction

**Bouchra && Laura**

**M2MLDS**

## 1-Comment fonctionne un SGAN ?

1- SGAN est un GAN dont le discriminateur est un classificateur multi-classe contrairement au discriminateur d'un GAN standard qui apprend à distinguer entre deux classes : Réelle et fausse.

2- Le discriminateur SGAN apprend à classifier  $N+1$  classes., ou  $N$  est le nombre de classe d'ensemble d'entraînement et une classe pour les images générées par le générateur.

3- Le but de SGAN Generator est le même que dans le GAN origine, il prend un vecteur aléatoires et produit de faux exemples.

4- SGAN Discriminator reçoit trois types d'entrées : des faux exemples produits par le générateur  $X^*$ , des vraies exemples sans étiquettes de l'ensemble de données d'apprentissage  $X$  et des vrais exemples avec des étiquettes de l'ensemble d'apprentissage  $(X,y)$ , où  $y$  est le label. Dans ce cas le discriminateur s'entraîne pour classer correctement ces trois types de données.

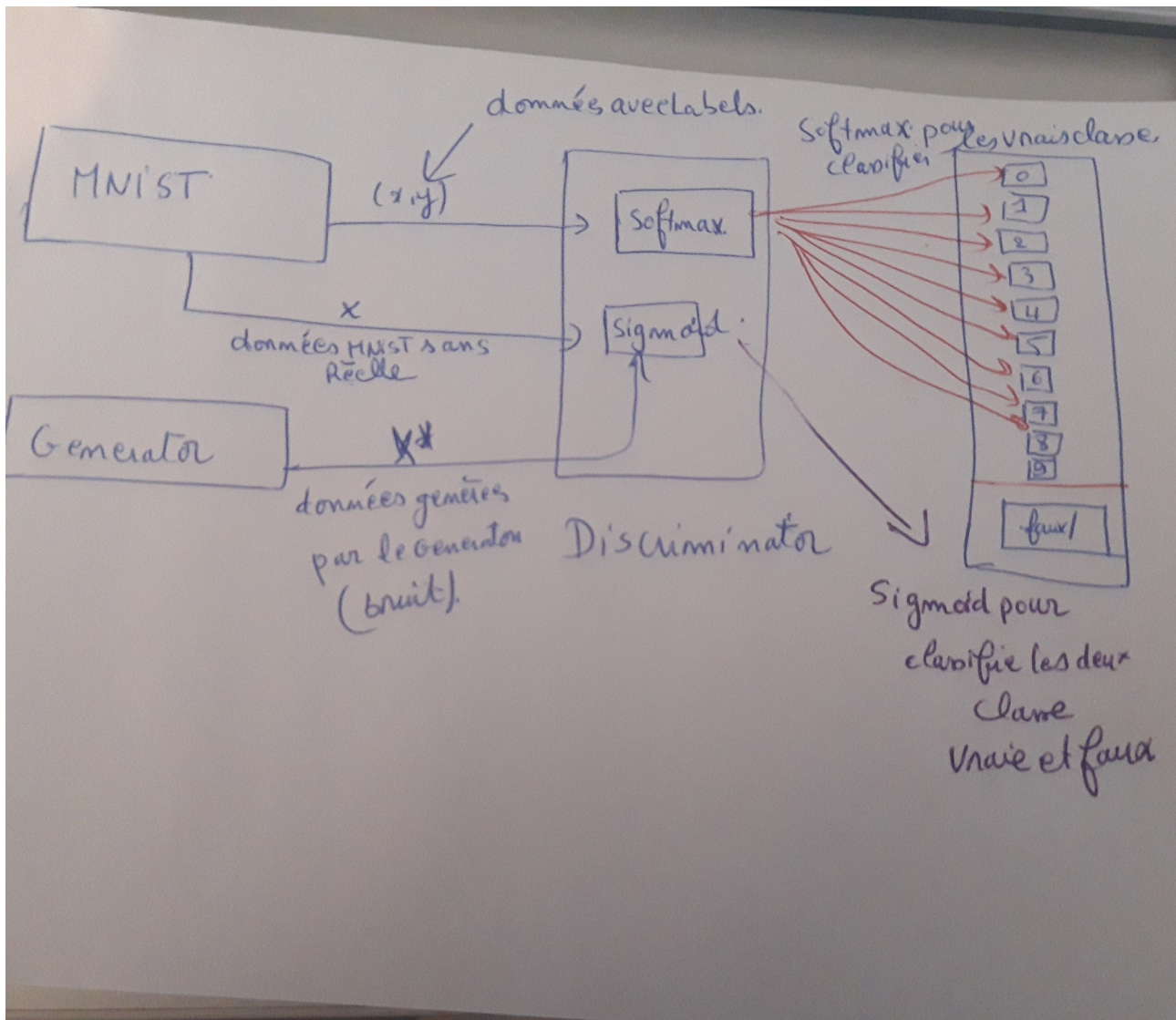
5-Pour un GAN standard, nous entraînons le Discriminateur en calculant la perte pour  $D(X)$  et  $D(X^*)$  et en rétropropagant la perte totale pour mettre à jour les paramètres entraînaibles du Discriminateur afin de minimiser la perte. Le générateur est formé en rétropropagant la perte du discriminateur pour  $D(X^*)$ , en cherchant à le maximiser, de sorte que les faux exemples qu'il produit sont mal classés comme réels.

6- Pour SGAN, en plus de  $D(x)$  et  $D(x^*)$ , il faut également calculer la perte pour les exemples de formation supervisée:  $D((x, y))$ . Ces pertes correspondent au double objectif d'apprentissage auquel le discriminateur SGAN doit s'entraîner : distinguer les vrais exemples des faux tout en apprenant à classer les vraies classes( $N$ ) : la perte supervisée et la perte non supervisée.

## 2- Expérimentation sur MNIST

Entraîner un SGAN qui apprend à classifier MNIST en utilisant 100 labels.

## 3- Architecture



#### 4-les fonctions implémentées :

**Pour la Class Dataset** : contient les fonction pour les opérations nécessaire pour les données : pre-processing des données- sélection les images avec labels/ sans labels- récupérer les données testes et les données entraînement

#### **les fonction de la Class SGAN :**

**build\_generator** : Création d'un générateur. Il prend comme entré un vecteur qui le transforme à une image 28x28x1

**build\_discriminator\_net** : implémentation des discriminateur standard . Le modèle termine par une Full couche connectée avec 10 neurones.

**build\_discriminator\_supervised** : Création de la partie supervisée de modèle discriminateur c'est à dire la classification multiclasse(softmax)

**build\_discriminator\_unsupervised** : Création de la partie non supervisée de modèle discriminateur c'est à dire la classification binaire

**combined** : Combine les deux modèles : Discriminateur et générateur.

**Train** :

1- Entraîner Discriminateur supervisé :

1-1 Sélectionner aléatoirement d'exemple réelles étiquetés

1-2 Minimisation de la perte de classification multiclasse.

2- Entraîner Discriminateur non supervisé

2-1 Sélectionner aléatoirement d'exemple réelles non étiquetés

2- 2 Minimisation de la perte de la classification binaire

3- Entraîner le générateur