

Generative Adversial Network

Balsan Thibault, Carvaillo Thomas, L'archevêque Valentin

Résumé

Intro ici

Partie ? - Modélisation mathématique

Dans cette partie, nous allons apporter les divers éléments mathématiques de la méthode ; en modélisant les perceptrons multicouches comme des distributions de probabilité, et en explicitant une méthode de construction optimale des perceptrons.

Le perceptron multicouche générateur G (resp. le discriminateur D) sera ici modélisé par une fonction différentiable $G_{\theta_g}(z)$ (resp. $D_{\theta_d}(x)$) ; où $z \sim p(z)$ est le bruit donné en entrée du générateur et $x = G_{\theta_g}(z)$. Nous dénoterons par p_{data} la distribution de probabilité de l'échantillon originel, et p_g la distribution de probabilité de l'échantillon généré par G . Le but de ce GAN étant ainsi la convergence de p_{data} vers p_g . Ici, $D_{\theta_d}(x)$ ne retournera pas une valeur binaire, mais un scalaire compris entre 0 et 1, représentant la probabilité que x soit ($D_{\theta_d}(x) = 1$) ou non ($D_{\theta_d}(x) = 0$) généré par le générateur.

Avant de présenter la formalisation du problème, nous allons évoquer un cas élémentaire, en émettant l'hypothèse forte que la distribution p_{data} est connue.

Proposition 1 (Optimalité de D) Soit G_{θ_g} un générateur fixe, alors le discriminateur optimal $D_{\theta_d}^*(x)$ est défini par

$$D_{\theta_d}^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}$$

Une démonstration est présentée en annexe.

Comme nous l'avons vu, il s'agit de maximiser la probabilité que D ait raison, i.e. maximiser $D_{\theta_d}(x)$, et, dans un même temps, que G trompe D , i.e. de minimiser $D_{\theta_d}(G_{\theta_g}(z))$. Ceci peut se réécrire sous le problème d'optimisation suivant :

$$\min_{G_{\theta_g}} \left(\max_{D_{\theta_d}} \left(\mathbb{E}_{x \sim p_{data}} \left[\log \left(D_{\theta_d}(x) \right) \right] + \mathbb{E}_{z \sim p_z} \left[\log \left(1 - D_{\theta_d}(G_{\theta_g}(z)) \right) \right] \right) \right)$$

Ce problème se résout de manière computationnelle, à l'aide d'une descente de gradient pour le problème de minimisation et d'une ascension de gradient pour celui de maximisation.

Algorithm 1 text

1: