Par Florian Coustures

Modélisation générative basée sur des scores grâce aux équations différentielles stochastiques

Dans ce travail nous résumerons les travaux présentés par Yang Song et al. la conférence ICLR 2021, avant d'en faire une application à un autre jeu de données. Nous présenterons ainsi leur méthode basée sur deux étapes distinctes : La création du bruit à partir des données, puis la modélisation générative à partir de ce bruit créé. Pour générer les échantillons, nous détaillerons les différences entre les 3 méthodes proposées par les auteurs. Enfin, nous nous intéresserons à la performance du modèle proposé ainsi qu'à ses différents champs d'application.

I) Résumé des travaux

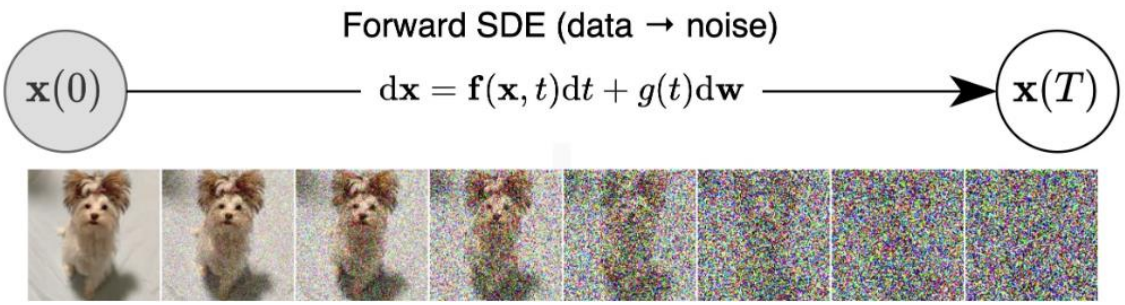
1) Bruiter puis générer les données

La méthode proposée de modélisation générative repose sur deux phases indissociables, soit une phase de corruption de la donnée par création du bruit en fonction du temps puis une phase d'inversion de cette corruption (toujours en fonction du temps) pour recréer la donnée. La subtilité du processus étant l'estimation d'un score en tout temps, ce qui est essentiel lors de la phase d'inversion. Pour ce faire, les auteurs utilisent deux modèles de scoring soit le *Score matching with Langevin dynamics* (SMLD) et le *Denoising diffusion probabilistic modeling* (DDPM). Ces modèles provenant de la littérature se sont avérés très utiles pour notre sujet d'études, soit la génération d'images.

Les auteurs ajoutent cependant certaines spécificités par rapport aux travaux existants pour créer une méthode complète unique. Notamment, plutôt que de considérer un nombre fini de perturbations des données ils préfèrent un processus continu ne reposant sur aucun paramètre. L'injection progressive du bruit prend ainsi la forme d'un processus stochastique continu dans le temps :

$$dx = f(x,t)dt + g(t)dw$$

Dans cette équation $f(.,t)$ est une fonction à valeur vectorielle représentant la déviation, $g(t)$ est une fonction à valeur réelle pour la diffusion, w est un mouvement brownien standard et ainsi dw est un bruit blanc. Si l'on applique cela à une image, cela donne donc le résultat suivant :



Comme nous connaissons la façon dont a été bruité la donnée, nous pouvons inverser le processus. La fonction d'inversion du bruit s'écrit alors ainsi :

$$dx = [\, f(x,t)dt - g^2(t)dw\, \nabla_x \log p_t(x)\,]\, dt + g(t)d\bar{w}$$

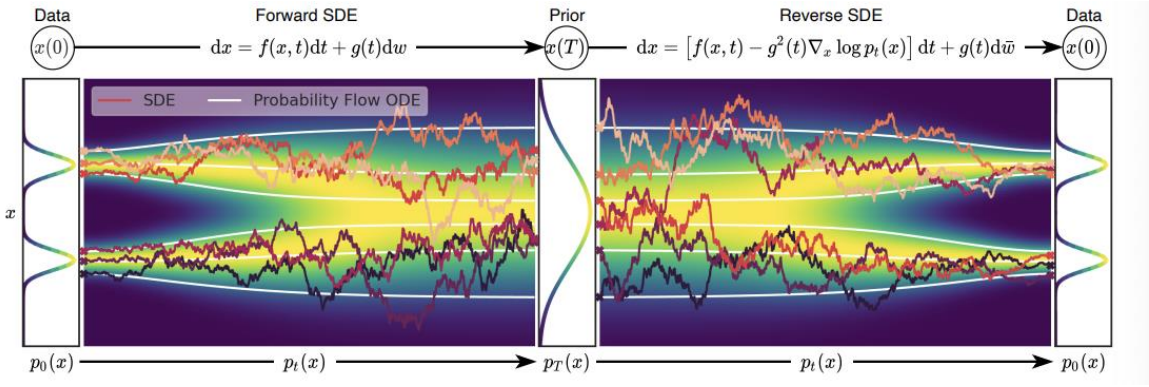
Avec $\bar{w}$ un processus de Wiener pour le temps retournant de T à 0 et dt un pas de temps négatif très petit. La fonction de score ($\nabla_x \log p_t(x)$) est alors estimée à partir d'un réseau de neurones dépendant du temps. Il est alors possible de créer des échantillons en utilisant des méthodes de résolution d'équations différentielles stochastiques (EDS).

2) Résoudre l'inversion de l'EDS

Les auteurs proposent trois méthodes principales de résolution de l'EDS. La première est une procédure numérique simple basée sur la méthode **Euler-Maruyama**. Cette méthode transforme la résolution de l'EDS en problème discret en utilisant un nombre fini, négatif et très petit de sauts dans le temps. La fonction est alors ajustée à l'aide de bruits Gaussiens jusqu'à ce que le temps soit à nouveau très proche de 0.

La deuxième approche est basée sur une méthode de Monte-Carlo par chaînes de Markov (MCMC), créant des échantillons dits **Prédicteurs-Correcteurs.** Le *prédicteur* pouvant être n'importe quelle méthode de résolution numérique permettant d'obtenir un nouvel échantillon à partir d'un échantillon existant (en faisant un saut dans le temps). Il est alors possible d'utiliser une procédure de MCMC (comme la dynamique de Langevin) dans le rôle du *correcteur*.

Enfin, la dernière méthode de résolution est basée sur la conversion de l'EDS en équation différentielle ordinaire (EDO), ce qui permet d'obtenir un **flux de probabilité EDO.** La trajectoire de distribution obtenue en résolvant le flux de probabilité EDO suivra la même trajectoire que celle de l'EDS, suivant l'image suivante :

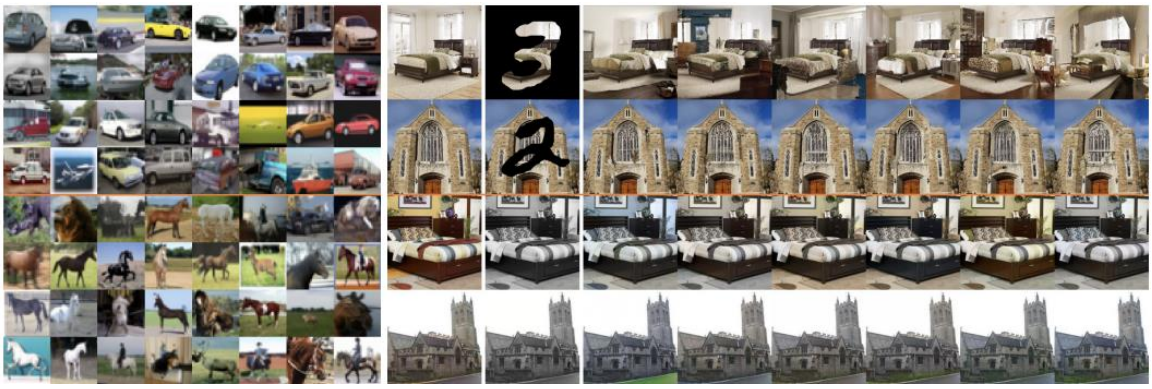


Cette dernière méthode a de nombreux avantages puisqu'elle s'intègre très bien à des réseaux de neurones. Surtout elle permet de calculer le logarithme de la vraisemblance, ce qui est très utile pour la comparaison des résultats aux modèles déjà existants.

3) Résultats et champs d'application

Pour apprécier la performance de leurs méthodes au regard des modèles existants les auteurs ont testés leur modélisation générative sur le jeu de données CIFAR-10. Les résultats sont déjà intéressants avec la méthode Euler-Maruyama mais deviennent les meilleurs selon les mesures choisies avec les autres méthodes de résolution. Notamment avec la méthode Prédicteur-Correcteur, ils battent le meilleur modèle à ce jour (StyleGan2 + ADA) si l'on prend la mesure FID ou le score Inception. Les performances sont d'autant plus remarquables avec le flux de probabilité EDO où là encore ils établissent une nouvelle marque de référence. Sur les images CIFAR-10 ils obtiennent ainsi la meilleure performance (en regardant la log-vraisemblance), et ont des résultats comparables aux meilleurs processus autorégressifs sur la base de données ImageNet.

Ces résultats impressionnants sont applicables à toute une série de problématiques de générations d'images. Les auteurs en retiennent notamment trois pour lesquelles ils démontrent l'utilité de leurs travaux. Tout d'abord la génération conditionnelle à une classe définie (Voir partie gauche de l'image suivante avec les voitures et chevaux). Ensuite, l'imputation d'images, avec ici des cas appliqués à des données manquantes (Deux lignes supérieurs de la partie droite). Pour finir, la colorisation d'images, ici appliqué à des objets et monuments (Deux lignes inférieurs de la partie droite) mais qui est aussi possible pour d'autres types d'images comme les visages.



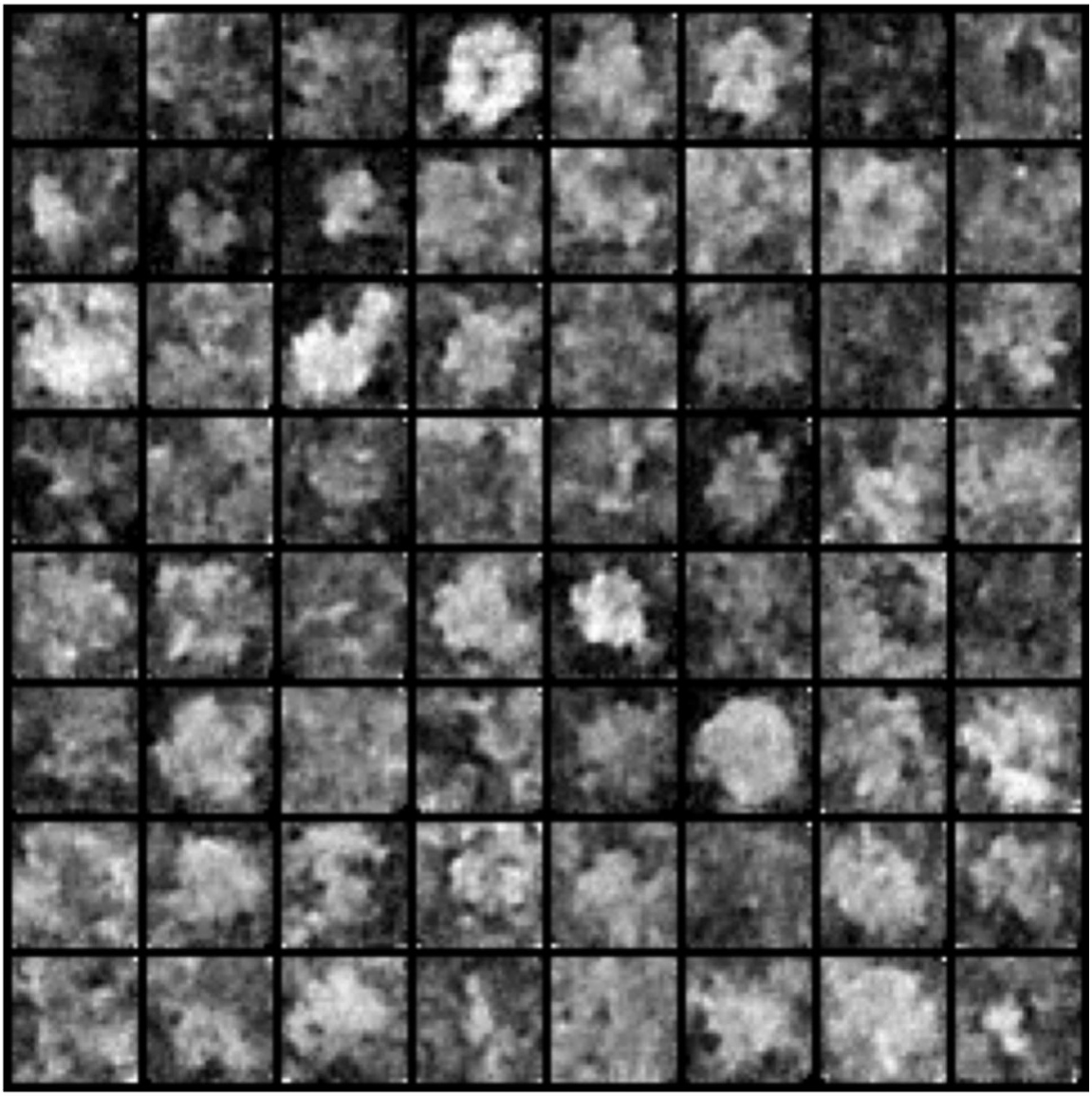
II) Utilisation des travaux

Pour utiliser les travaux de Yang et al. nous nous servons d'un jeu de données de 102 types de fleurs contenant près de 8189 images de fleurs : https://www.robots.ox.ac.uk/~vgg/data/flowers/102/index.html.

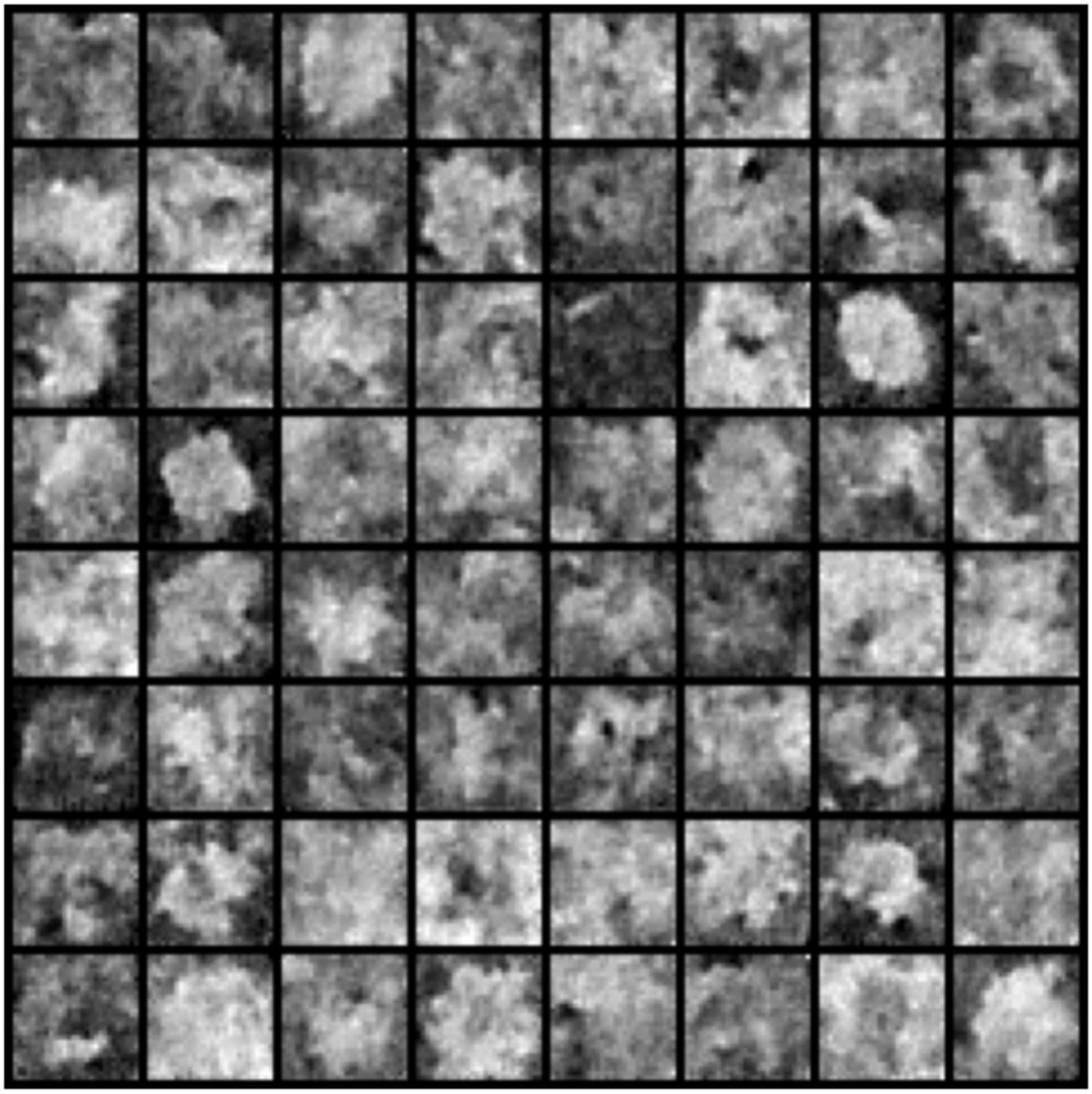
Voici l'exemple d'une fleur après application du filtre en noir et blanc :

Nous générons par la suite les images en suivant les méthodes proposées entraînant le modèle avec 200 epochs (fichier disponible sur le github), des batch de 32 et un learning rate de 1 x 10^{-4}.

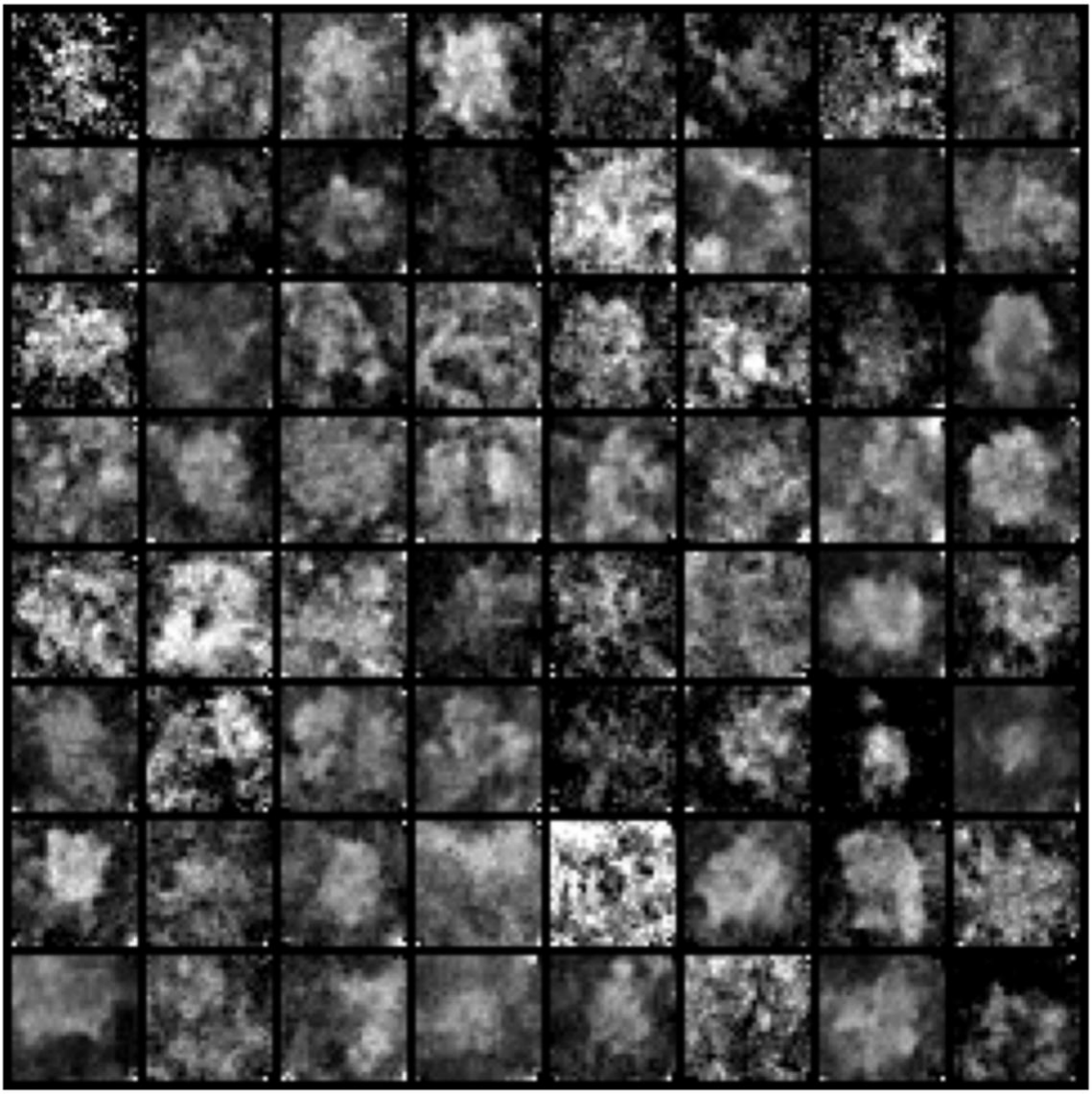
Avec la méthode Euler-Maruyama et un choix de 2000 étapes :



Avec la méthode Predicteur-Correcteur et un choix de 2 000 étapes et un noise ratio de 0,15 :



Avec la méthode du flux de probabilité EDO :



Finalement, nous pouvons observer que certaines fleurs apparaissent plutôt clairement à la fin du processus. Les trois méthodes donnent des résultats assez différents et le flux de probabilité EDO apparaît donner la meilleure performance. Les résultats sont cependant assez hétérogènes à cause du filtrage en noir et blanc. Certaines fleurs du jeu de données initial sont difficilement distinguables sans couleur, ce qui rend la génération d'images plus complexes.

Conclusion :

Les travaux de Yang et al. basée sur une méthode originale de générations d'images à partir du bruit ont permis de nouvelles avancées. Leur méthode prometteuse pourrait permettre des améliorations diverses (génération en fonction d'une classe conditionnelle, imputation d'image, colorisation). S'ils restent moins rapides pour générer des échantillons que des modèles GANs, nous pouvons apprécier la qualité des résultats, comme cette colorisation du visage d'Abraham Lincoln.