Une approche innovante pour l'analyse de séries bruitées à l'aide de l'algorithme SINDy et de la différentiation automatique

March 2023

Table des matières

1 Introduction

La problématique traitée par cet article est l'**identification de modèles dynamiques non linéaires à partir de données expérimentales bruitées**. Il s'agit d'une tâche importante en sciences et ingénierie pour comprendre le comportement de systèmes complexes et concevoir des contrôleurs efficaces.

Plus précisément, nous proposons une méthode basée sur la différenciation automatique et l'algorithme SINDy (Sparse Identification of Nonlinear Dynamics) pour estimer les équations différentielles qui décrivent le comportement d'un système dynamique à partir de mesures bruitées.

La **différenciation automatique** est une technique mathématique qui permet de calculer automatiquement les dérivées d'une fonction en utilisant des règles de dérivation chainée. Contrairement à la différenciation mathématique qui utilise des formules analytiques pour calculer les dérivées, et à la différenciation numérique qui approxime les dérivées à partir de la discrétisation de la fonction, la différenciation automatique permet d'obtenir des valeurs précises des dérivées d'une fonction en évitant les erreurs d'approximation et en garantissant la précision numérique. Le principe mathématique repose en général sur la règle de la chaîne.

Historiquement, la différenciation automatique a été développée dans les années **1950** pour les applications en aérospatiale, où les calculs des trajectoires de missiles et de satellites nécessitaient des calculs précis des dérivées. Depuis, la différenciation automatique a été utilisée dans divers domaines scientifiques et techniques, notamment pour l'optimisation, la simulation numérique, la modélisation et la commande de systèmes dynamiques.

L'algorithme **SINDy**, quant à lui, est une méthode d'apprentissage automatique qui permet d'identifier les termes non linéaires pertinents dans un système dynamique en utilisant une régularisation L1 pour encourager la parcimonie.

Ainsi au sein d'un cadre d'optimisation, le bruit peut être séparé du signal, résultant en une architecture qui est environ deux fois plus robuste au bruit que les méthodes de pointe. Nous montrerons que la méthode peut apprendre une diversité de probabilitédistributions pour le bruit de mesure, y compris gaussien, uniforme, gamma et Rayleigh. distributions La méthode consiste en :

- débruiter les données
- apprendre et paramétrer la distribution de probabilité de bruit
- identifier la **parcimonie** , c'est à dire le modèles le plus "simple" et le plus explicatif

Concrètement, on applique la méthode d'apprentissage de cette manière :

- Collecte des données : Tout d'abord, il est nécessaire de collecter des données du système dynamique que vous souhaitez modéliser. Cela peut se faire à l'aide de capteurs, d'expériences ou de simulations.
- Sélection des variables : Ensuite, il faut sélectionner les variables qui interviennent dans le système. Ces variables peuvent être des variables mesurées ou des variables dérivées qui peuvent être calculées à partir des variables mesurées.
- Construction de la matrice de données : En utilisant les données collectées, on construit une **matrice de données X** qui contient les valeurs des variables sélectionnées à chaque instant de mesure. Cette matrice aura une taille N x M, où N est le nombre d'observations et M est le nombre de variables sélectionnées.

$$\frac{dx(t)}{dt} = f(x(t))$$

- Calcul des dérivées : La prochaine étape consiste à calculer les dérivées temporelles des variables, qui sont nécessaires pour déterminer les équations différentielles sous-jacentes. Pour cela, on peut utiliser une méthode de différences finies ou une méthode de dérivation automatique. On les stockera dans la **matrice Y des dérivées partielles**.
- Résolution d'un problème d'optimisation : Une fois que les dérivées temporelles sont calculées, on utilise une méthode d'optimisation sparse pour trouver les termes non linéaires qui doivent figurer dans les équations différentielles sous-jacentes. Cela consiste à minimiser une fonction objectif qui combine l'erreur entre les données et les prédictions du modèle, et une pénalité sur la norme L1 des coefficients de la matrice des dérivées partielles.
- Validation du modèle : Enfin, le modèle SINDy obtenu doit être validé en le testant sur de nouvelles données ou en comparant ses prédictions à celles d'autres modèles existants.

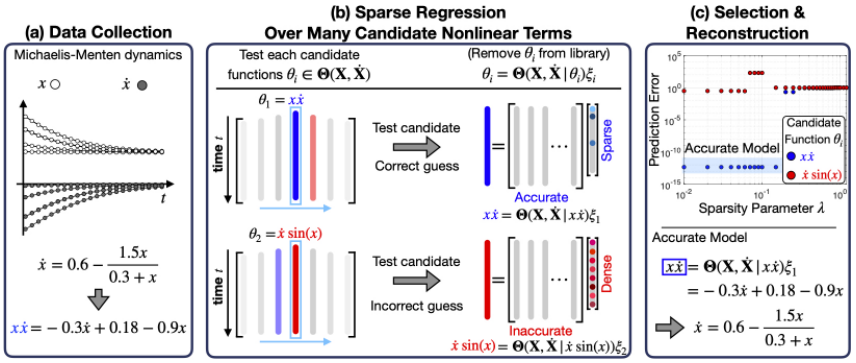
Soit la matrice des coefficients qui relie les variables de X, tels que X = X. L'objectif de SINDy est de trouver la matrice qui minimise l'expression suivante :

$$\min_{\Theta} \|Y - X\Theta\|^2 + \lambda \|\Theta\|_1$$

où Y est le vecteur des observations, X est la matrice des variables explicatives, Θ est le vecteur des coefficients à estimer, $\|\cdot\|^2$ est la norme euclidienne au carré, $\|\cdot\|_1$ est la norme L_1, λ est le paramètre de régularisation qui contrôle le compromis entre la qualité de l'ajustement et la parcimonie du modèle, N est le nombre de points de données dans la série $X(t)$.

où est un paramètre de régularisation qui contrôle la sparsité de . La solution optimale pour est obtenue en utilisant des méthodes de programmation convexe, telles que Lasso ou Elastic Net.

Le principe peut être résumé sur l'image suivante (https://github.com/dynamicslab/SINDy-PI/blob/master/Images/DL_SINDy.jpg)



2 Algorithme de la nouvelle méthode

La méthode proposée dans l'article utilise des techniques d'estimation de densité de probabilité pour extraire les distributions de probabilité du bruit dans les données. Mathématiquement, cela peut être décrit comme suit :

Soit X(t) la série de données expérimentales collectées à des instants de temps t_1, t_2, ..., t_N. Cette série de données peut être décrite comme une combinaison de l'état véritable x(t) et d'un bruit n(t) :

X(t) = x(t) + n(t)

On a ainsi **initialisé** dans notre algorithme la valeur :
$\Xi = \text{SINDy}(X, \Theta(X), \lambda)$

Ensuite, on cherche **en itérant** à **optimiser** la gestion du bruit en minimisant la distance :

$$\mathcal{E}_d = |\hat{X}' - \Theta(\hat{X})\Xi|_2^2 \tag{1}$$

Plus précisément, on cherche à trouver la fonction f(x) qui représente la dynamique du système en résolvant l'équation différentielle suivante :

$x(t+1) = x(t) + \delta t \cdot f(x(t))$

où x(t) représente l'état du système au temps t, f(x(t)) représente la dynamique du système et t est l'intervalle de temps entre deux mesures successives. L'objectif est de trouver une expression analytique de f(x) en utilisant les données bruitées disponibles.

Dans l'article, on utilise une méthode de **Runge Kutta à l'ordre 4**.

La méthode SINDy utilise une optimisation parcimonieuse pour trouver l'expression la plus simple de f(x) qui explique les données. Dans le cas présent, on cherche à minimiser la fonction de coût suivante :

$L = ||x(t+1) - x(t) - \Delta t \cdot f(x(t))||^2 + \lambda||\Theta||_1$

où est un hyperparamètre de régularisation et représente les coefficients de f(x) dans une base de fonctions. La **norme L1 est utilisée comme pénalité** pour favoriser la parcimonie de la solution.

Pour résoudre cette optimisation, l'algorithme de **descente de gradient** qui permet de trouver la solution optimale tout en imposant une contrainte de parcimonie. Les coefficients trouvés par l'optimisation représentent ainsi les paramètres optimaux de f(x) pour réduire le bruit dans les données.

Une fois la densité de probabilité du bruit estimée, elle peut être utilisée pour ajuster le modèle SINDy de manière à prendre en compte l'impact du bruit sur les données expérimentales. Cela permet de fournir des modèles plus précis et plus robustes.

3 Résultats obtenus

3.1 Nouvelle méthode vs réseaux de neurones

Plusieurs exemples sont traités par notre étude comme l'oscillateur de Van der Pool, l'attracteur de Rössler ou de Lorentz.

Pour le système chaotique décrit par l'attracteur de Lorentz, les fonctions de la matrices de départ sont définies par les équations physiques :

$$\frac{dx}{dt} = \sigma(y - x) \quad \frac{dy}{dt} = x(\rho - z) - y \quad \frac{dz}{dt} = xy - \beta z$$

où σ, ρ et β sont des constantes positives.

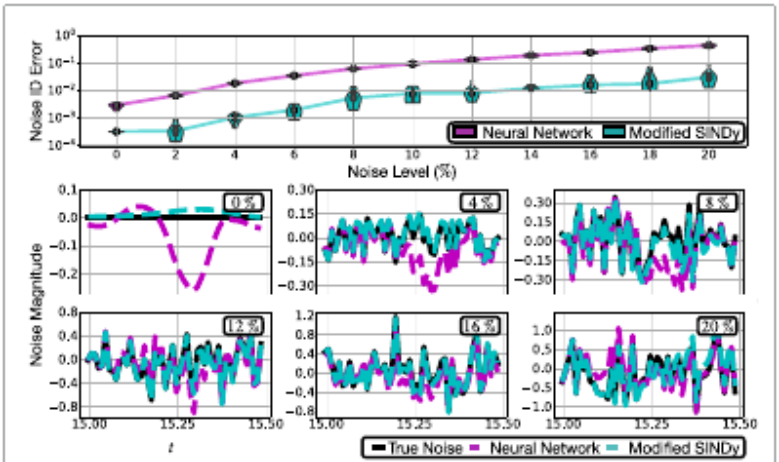


Figure 2. Top: Noise identification error of modified SINDy (labeled as SINDy) and NN denoising approach by Rudy *et al* [32]. The black circle represents the median of ten runs while the violin shape represents the distribution of error. The modified SINDy approach shows better noise identification error. Bottom: Comparison between the average noise applied to the Lorenz system and the noise identified by the two approaches. As shown on the left, both approaches can not produce the correct zero noise result when no noise is applied, which happens since there is a tiny difference between the learned dynamics and true dynamics.

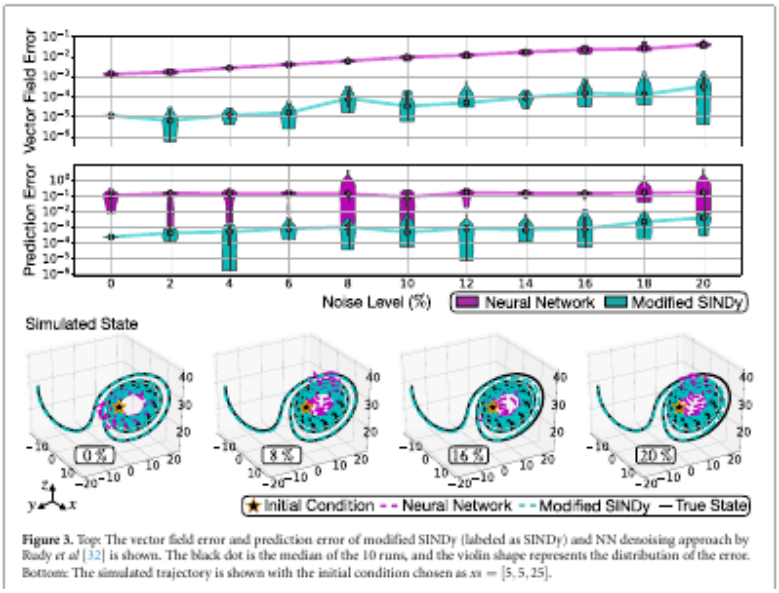


Figure 3. Top: The vector field error and prediction error of modified SINDy (labeled as SINDy) and NN denoising approach by Rudy *et al* [32] is shown. The black dot is the median of the 10 runs, and the violin shape represents the distribution of the error. Bottom: The simulated trajectory is shown with the initial condition chosen as $x_0 = [5, 5, 25]$.

On définit l'erreur d'identification du bruit :

$$E_N = \frac{1}{m} \sum_{i=1}^{m} \|n_i - \hat{n}_i\|_2^2$$

Ainsi que le vecteur d'erreur :

$$E_f = \frac{1}{m}\sum_{i=1}^{m} \frac{\|f_i\|^2_{(x-\bar{x}i)}}{\left\| \frac{f_i}{\|f_i\|(x-\bar{x}_i)} - n_i \right\|_2^2} \quad (14)$$

- Le SINDy modifié montre une **meilleure erreur d'identification du bruit**.
- L'erreur de prédiction du SINDy modifié est aussi meilleure et le traitement plus **performant** que l'approche par réseau de neurone.

3.2 Interprétation des résultats

Le paramètre de seuil est le paramètre le plus important à régler dans notre méthode SINDy modifiée. Le paramètre va déterminer la parcimonie de la structure du modèle.

- Si la valeur de est trop petit, la contrainte de **parcimonie** ne sera pas assez forte pour imposer le bon modèle à trouver.
- Si la valeur de est trop grande, les termes corrects peut être éliminé à tort et on obtiendra un **modèle erroné**. Si la structure du modèle est erronée, il y aura une énorme différence entre le bruit identifié N et le bruit réel N.

Dans le cas de Lorentz, les résultats ont montré que la méthode SINDy peut identifier les termes de plus haut ordre du modèle même avec des niveaux de bruit élevés. Le paramètre lambda affecte la qualité du modèle extrait, et il est important de trouver la bonne valeur de lambda pour obtenir les meilleures performances.

Dans le cas de Lotka-Volterra, la méthode SINDy a montré qu'elle peut extraire avec précision le modèle sous-jacent même avec un bruit important. Le paramètre lambda a également un impact significatif sur les performances de la méthode. Des valeurs de lambda plus élevées ont tendance à produire des modèles plus simples mais moins précis, tandis que des valeurs de lambda plus faibles peuvent conduire à des modèles plus complexes mais également plus précis.

4 Conclusion

Nous avons présenté une **nouvelle méthode** pour identifier les équations différentielles sous-jacentes à partir de données expérimentales, tout en **extrayant les distributions de probabilité du bruit** dans les données. Cette méthode combine des techniques de différenciation automatique avec des méthodes d'estimation de densité de probabilité pour extraire des informations plus complètes des données expérimentales.

En comparant les résultats de cette méthode avec les résultats de la méthode SINDy et des réseaux de neurones, les auteurs ont constaté que leur méthode **fournissait des modèles plus précis** et plus faciles à interpréter que les autres méthodes. En particulier, leur méthode était plus efficace pour identifier les termes non linéaires dans les équations différentielles, ce qui conduisait à des modèles plus simples et plus facilement interprétables.