EA-Sindy

Alois D'uston, ald6fd

April 2, 2023

Model defenition:

Base Autoencoder Sindy begins with data matrixes $X, \dot{X} \in \mathbb{R}^{m\times D}$.

Each column $X_i \sim \mathcal{X}$. Derivatives $\dot{X}_i$ in $\dot{X}$ are computed numerically.

We assume that $x \sim \mathcal{X}$ admits some lower dimensional representation in latent space $\mathcal{Z}$. We further assume that the dynamical system $\dot{x} = f(x)$ can be represented in sparse fashion in $\mathcal{Z}$. Concretely this means we fit model

I. $x = \varphi^{-1}\varphi(x) = \varphi^{-1}(z)$, II. $(\frac{d}{dx}\varphi)\dot{x} = \frac{d}{dt}\varphi(x) = \dot{z} = \Theta(z)\Xi = \Theta(\varphi(x))\Xi$,

III. $\dot{x} = \frac{d}{dt}\varphi^{-1}\varphi(x) = (\frac{d}{dz}\varphi^{-1})(\frac{d}{dt}\varphi(x)) = (\frac{d}{dz}\varphi^{-1})(\Theta(\varphi(x))\Xi)$, where

i. $\varphi : \mathbb{R}^D \to \mathbb{R}^d$, $\varphi^{-1} : \mathbb{R}^d \to \mathbb{R}^D$ are encoder decoder neural nets.

ii. $\Theta : \mathbb{R}^d \to \mathbb{R}^p$ evaluates function library(consisting of p funcs) on $\varphi(x)$.

iii. $\Xi \in \mathbb{R}^{d\times p}$ is a matrix of coefficients associated to each term in $\Theta(\varphi(x))$, which we take to be sparse.

We fit this model using gradient descent with loss function:

$\mathcal{L}(\varphi, \varphi^{-1}, \Xi; x, \dot{x}) = \mathcal{L}_{\text{encode}}(\varphi, \varphi^{-1}; x) + \mathcal{L}_{\text{fit}}(\varphi, \varphi^{-1}, \Xi; x, \dot{x}) + \mathcal{L}_{\text{reg}}(\Xi)$

$\mathcal{L}_{\text{encode}}$ enforces that $(\varphi^{-1} \circ \varphi)\big|_{\mathcal{X}} \approx I$, $\mathcal{L}_{\text{fit}}$ enforces conditions II,III and $\mathcal{L}_{\text{reg}}$ promotes sparsity in coefficients Ξ. Usually use $\mathcal{L}_{\text{reg}}(\Xi) = \lambda_{\text{reg}}\|\Xi\|_1$

L_1 regularization results in Ξ matrix where lots of coefficients are close to zero. Our prior assumption is that only a few coefficients are nonzero.

We use coefficient mask Λ to enforce coherence to this assumption.
At epoch k of training we set $\Lambda_{ij}^{(k)} = \mathbb{1}\{(\Lambda^{(k-1)} \odot \Xi^{(k-1)})_{ij} \approx 0\}$.

In ensemble autoencoder sindy we split our training data $X, \dot{X}$ into b bags $\{X^{(i)}, \dot{X}^{(i)}\}_{i=1}^{b}$ where $X^{(i)}, \dot{X}^{(i)} \in \mathbb{R}^{q \times D}$ are sampled from the training samples $X, \dot{X}$ with replacement. We consider coefficient tensor $\Xi^{[1:b]} \in \mathbb{R}^{b \times d \times p}$, where $\Xi^{[1:b]}_{[i,:,:]} = \Xi^{(i)}$ corresponds to bag $X^{(i)}, \dot{X}^{(i)}$ of the data. In other words, $\Xi^{(i)}$ is used to fit $\frac{d}{dt}\varphi(x)$ to $\dot{x}$ for $x, \dot{x} \in X^{(i)}, \dot{X}^{(i)}$.
As with regular autoencoder sindy, in training, we fit the model
I. $x = \varphi^{-1}\varphi(x) = \varphi^{-1}(z)$, II. $\frac{d}{dt}\varphi(x) = \Theta(\varphi(x)) \sum_i \Xi^{(i)} \mathbb{1}\{x \in X^{(i)}\})$
III. $\dot{x} = (\frac{d}{dz}\varphi^{-1})(\Theta(\varphi(x)) \sum_i \Xi^{(i)} \mathbb{1}\{x \in X^{(i)}\})$
We do so via gradient descent using loss function:
$\mathcal{L}(\varphi, \varphi^{-1}, \Xi^{[1:b]}; x, \dot{x}) = \mathcal{L}_{\text{encode}}(\varphi, \varphi^{-1}; x) + \mathcal{L}_{\text{fit}}(\varphi, \varphi^{-1}, \Xi^{[1:b]}; x, \dot{x}) + \mathcal{L}_{\text{reg}}(\Xi^{[1:b]})$
$\mathcal{L}_{\text{encode}}$ is unchanged, enforces that $(\varphi^{-1} \circ \varphi)\big|_{\mathcal{X}} \approx I$, $\mathcal{L}_{\text{fit}}$ enforces II,III and $\mathcal{L}_{\text{reg}}$ promotes sparsity in the average of coefficients $\Xi^{(1)}, \ldots \Xi^{(b)}$
Ie it promotes sparsity in $\tilde{\Xi} = \frac{1}{b}\sum_i \Xi^{(i)}$. We use $\mathcal{L}_{\text{reg}}(\Xi^{[1:b]}) = \lambda_{\text{reg}} \|\frac{1}{b}\sum_i \Xi^{(i)}\|_1$
The final output of the model, used in testing, is the avg coefficient matrix $\tilde{\Xi} = \frac{1}{b}\sum_i \Xi^{(i)}$. We again use a coefficient mask Λ to enforce our prior assumption that only a few coefficients of $\tilde{\Xi}$ are nonzero. At epoch k of training we set $\Lambda_{ij}^{(k)} = \mathbb{1}\{\frac{1}{b}|\{h : (\Lambda^{(k-1)} \odot \Xi^{(h),(k-1)})_{ij} \approx 0\}| > p_{\text{tol}}\}$.
This procedure is known is stability selection. It works by finding all those coefficients Ξ_{ij} which aren't consistently activated(well seperated from 0) across data bags $\{X^{(i)}, \dot{X}^{(i)}\}_{i=1}^{b}$, and setting these coefficients to 0.

Results:

Base test:

Low data test:

Noise test:

Avg v Inclusion test:

Total reg v Avg reg test:

Research directions:

Implementation details: