EA-Sindy

Alois D'uston, ald6fd

April 4, 2023

Autoencoder Sindy:

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

Each column X_i of the data matrix is taken to come from some distribution with support on $\mathcal{X} \subset \mathbb{R}^D$. Derivatives $\dot{X}_i$ in data matrix $\dot{X}$ are computed numerically beforehand. We assume that data vectors $x \in \mathcal{X}$ admit lower dimensional representations in some latent space $\mathcal{Z}$. We further assume that the dynamical system $\dot{x} = f(x)$, mapped into $\mathcal{Z}$, can be represented as $\dot{z} = \Theta(z)\Xi$, where Θ is a function library evaluated on the coordinates of z and Ξ is sparse matrix of coefficients.

We further assume that the mappings $x \mapsto z$ and $z \mapsto x$ are differentiable.

Concretely, the above assumptions can be encoded as

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

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

ii. $\Theta : \mathbb{R}^d \to \mathbb{R}^p$ evaluates function library(containing p functions) 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 in the $\|\cdot\|_0$ sense.

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 $\varphi^{-1}\varphi(x) \approx x$, $\mathcal{L}_{\text{fit}}$ enforces conditions II,III and $\mathcal{L}_{\text{reg}}$ promotes sparsity in coefficients Ξ, typically using $\mathcal{L}_{\text{reg}}(\Xi) = \lambda_{\text{reg}}||\Xi||_1$. 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 exact coherence to this assumption.
At epoch k,of training we set $\Lambda^{(k)}_{ij} = \mathbb{1}\{(\Lambda^{(k-1)} \odot \Xi^{(k)})_{ij} \approx 0\}$ for all indexes $i = 1...d$, $j = 1...p$ of the to coefficient matrix. This mask has the effect of setting all coefficients in Ξ close to zero to exactly zero.

Ensemble Autoencoder Sindy:

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 uniformly from the training samples $X, \dot{X}$ with replacement. The idea will be to fit a sindy to each data bag, then in testing, to average over the outputs of all the models. This procedure should in principle produce a sindy predictor with less variance than the original. To do this 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 $x, \dot{x} \in X^{(i)}, \dot{X}^{(i)}$ per conditions II, III. In this framework our sindy predictor of the derivative $\dot{z}$ changes from $\Theta(\varphi(x))\Xi$ to $\Theta(\varphi(x))\sum_i \Xi^{(i)}\mathbb{1}\{x \in X^{(i)}\})$.
Hence, in training we fit the model
IV. $x = \varphi^{-1}\varphi(x)$, V. $\frac{d}{dx}\varphi(x)\dot{x} = \Theta(\varphi(x))\sum_i \Xi^{(i)}\mathbb{1}\{x \in X^{(i)}\})$
VI. $\dot{x} = (\frac{d}{dz}\varphi^{-1})(\Theta(\varphi(x))\sum_i \Xi^{(i)}\mathbb{1}\{x \in X^{(i)}\})$
where coefficient tensor $\Xi^{[1:b]}$ is taken to be sparse some appropriate way.
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, it still enforces that $\varphi^{-1}\varphi(x) \approx x$.
$\mathcal{L}_{\text{fit}}$ enforces V,VI, ie it enforces that for each i, $\Xi^{(i)}$ can be used to fit bag

$X^{(i)}$ to its corresponding numerically computed derivatives $\dot{X}^{(i)}$ per II, III. As mentioned $\mathcal{L}_{\text{reg}}$ promotes sparsity in $\Xi^{[1:b]}$. We may take this a number of ways. One approach is to promote sparsity in the average of the coefficients given $\tilde{\Xi} = \frac{1}{b}\sum_i \Xi^{(i)}$. To do this we use $\mathcal{L}_{\text{reg1}}(\Xi^{[1:b]}) = \lambda_{\text{reg}}\|\frac{1}{b}\sum_i \Xi^{(i)}\|_1$. The rationale here is that our final outputed predictor will be the bag average $\tilde{\Xi} = \frac{1}{b}\sum_i \Xi^{(i)}$, hence this is the object whose sparsity we should promote with regularization . Note, this procedure couples the the training of predictors $\Xi^{(i)}, \Xi^{(j)}$ for $i \neq j$ via the regularization term. $\Xi^{(i)}$ is incentivized to destructively interfere with terms in $\Xi^{(j)}$.

Generally speaking it is desirable to make models in an ensemble independent of one another. This observation motivates an alternative regularization scheme were we try to ensure that each of $\Xi^{(i)}$ are sparse in the $\|\cdot\|_0$ sense.(instead of the average being sparse). To do this we may use $\mathcal{L}_{\text{reg2}}(\Xi^{[1:b]}) = \lambda_{\text{reg}}\frac{1}{b}\sum_i \|\Xi^{(i)}\|_1$. Each of these regularizations has its pro and cons, which we discuss later in greater detail.

We again use a coefficient mask Λ to enforce our assumption that only a few coefficients of the bag average predictor $\tilde{\Xi} = \frac{1}{b}\sum_i \Xi^{(i)}$ are nonzero. We may achieve this in two ways. Firstly, at epoch k of training we may set $\Lambda_{ij}^{(k)} = \mathbb{1}\{\frac{1}{b}|\{h : (\Lambda^{(k-1)} \odot \Xi^{(h),(k)})_{ij} \approx 0\}| > p_{\text{tol}}\}$.

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. The reasoning here is that if a large proportion of the sub predictors $\{\Xi^{(i)}\}_{i=1}^{b}$ 'decide' that a coefficient is not important, then it is reasonable to set that coefficient to zero. This procedure implicitly assumes that these decisions are being made independently by the different sub-predictors. Hence this masking scheme is in theory more suited to the $\mathcal{L}_{\text{reg2}}$ regularizer which decouples the regularization of each sub model.

When sub-model regularization is coupled, and thus depedent, as with the $\mathcal{L}_{\text{reg1}}$ regularizer we may consider mask $\Lambda_{ij}^{(k)} = \mathbb{1}\{\Lambda^{(k-1)} \odot \frac{1}{b}\sum_h \Xi_{ij}^{(h),(k)} \approx 0\}$. This is the same masking scheme as in regular autoencoder sindy, with $\frac{1}{b}\sum_h \Xi_{ij}^{(h),(k)} = \tilde{\Xi}^{(h)}$ substituted for $\Xi^{(h)}$. One way to interpret this scheme is that sub predictors $\{\Xi^{(i)}\}_{i=1}^{b}$ cooperate to produce sparsity in $\tilde{\Xi}$ while preserving their own ability to fit their respective data bags $\{X^{(i)}, \dot{X}^{(i)}\}_{i=1}^{b}$. There is less freedom to manipulate the value of relevant/important coefficients than their is irrelevant ones. Hence the ensemble will preferentially destructively interfere to produce sparsity in $\tilde{\Xi}$ at those indexes i, j which are less important to the overall quality of the fit. This will result in the average $\frac{1}{b}\sum_h \Xi_{ij}^{(h),(k)}$ being well seperated from zero for relevant coefficients and close to zero for irrelevant coefficients, as to promote sparsity in $\tilde{\Xi}$. Note, we don't expect destructive interference for relevant coefficients. To see why, suppose $f(z) \in \Theta(z)$ is important coefficient for fitting $\dot{z}$. Further suppose for sub predictors, $\Xi^{(a)}, \Xi^{(b)}$, the coefficients α_a, α_b associated to f interfere destructively, ie $\alpha_a + \alpha_b \approx 0$.

Per the model setup we have $\dot{z} \approx \alpha_a f(z) + c_1^T g(z)$ and $\dot{z} \approx +\alpha_b f(z) + c_2^T g(z)$. Adding the two expressions we obtain $\dot{z} \approx \frac{1}{2}(c_1 + c_2)^T g(z)$, hence f is not essential to fitting $\dot{z}$. Even if the fit quality is the same, it might still be that representations $[\alpha_a \; c_1]$ or $[\alpha_b \; c_2]$ are 'better' than $[0 \; c_1 + c_2]$ in the sense that they have greater sparsity, but in that case the $\mathcal{L}_{\text{reg1}}$ regularizer would push the model in the direction of adopting the representation with the greatest sparsity. Hence in all cases it is reasonable for us to not worry about destructive interference among relevant coefficients in this masking scheme. We refer to this procedure as cooperative selection.

Implementation details:

Results:

Base test:

Low data test:

Noise test:

Avg v Inclusion test:

Total reg v Avg reg test:

Research directions: