Contents

[Initial attempt 2](#_Toc110331657)

[Looking at train and test data 4](#_Toc110331658)

[Ensemble & -pi with normal and fault 6](#_Toc110331659)

[Changing the library 11](#_Toc110331660)

[Combining data 12](#_Toc110331661)

# Initial attempt

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objective** | | Use SINDy to recover dynamics from the normal operation data (case western) | | |
| **Data** | **SINDy** | **Result** | **Output** | **Comment** |
| 0 hp data as training and 1 hp data as test | Ensemble | Straight line | training model  (x0)' = 0.002 x0 + -0.081 x0^2  Ensemble with replacement (V1)  (x0)' = 0.000  Ensemble model without replacement (V2)  (x0)' = 0.000 | Look at different SINDy method  Different training and test data ?  Filename: normal\_in\_ens\_sindy.py |
| Normal 0hp data | SINDy-PI  (PDE library, SINDy-PI optimiser) | Straight line | 1 = nan 1 + nan x0 + nan x0x0 + nan x0\_t + nan x0\_tt + nan x0x0\_t + nan x0x0x0\_t + nan x0x0\_tt + nan x0x0x0\_tt  x0 = 0.0124 1 + -1.6344 x0\_tt  x0x0 = 0.0052 1  x0\_t = 0.0000  x0\_tt = 0.0026 1 + -0.2224 x0  x0x0\_t = 0.0000  x0x0x0\_t = 0.0000  x0x0\_tt = -0.0012 1  x0x0x0\_tt = 0.0000 | Filename: normal\_sindypi  Model 8  Changing threshold - |
|  |  |  | 1 = nan 1 + nan x0 + nan x0x0 + nan x0\_t + nan x0\_tt + nan x0x0\_t + nan x0x0x0\_t + nan x0x0\_tt + nan x0x0x0\_tt  x0 = 0.0060 1 + 2.8206 x0x0 + -0.0725 x0\_t + -1.0406 x0\_tt + 3.7635 x0x0\_t + -2.2784 x0x0x0\_t + 8.1665 x0x0\_tt + -97.0856 x0x0x0\_tt  x0x0 = 0.0014 1 + 0.0280 x0 + 0.0078 x0\_t + 0.0343 x0\_tt + -0.0373 x0x0\_t + 3.3438 x0x0x0\_t + -2.6852 x0x0\_tt + 0.1333 x0x0x0\_tt  x0\_t = 0.0006 1 + -0.0324 x0 + 0.3424 x0x0 + -0.0701 x0\_tt + 0.3349 x0x0\_t + 73.0052 x0x0x0\_t + 1.3057 x0x0\_tt + -2.7058 x0x0x0\_tt  x0\_tt = 0.0023 1 + -0.1818 x0 + 0.5975 x0x0 + -0.0274 x0\_t + 1.3609 x0x0\_t + -1.8159 x0x0x0\_t + 1.8965 x0x0\_tt + 24.3546 x0x0x0\_tt  x0x0\_t = -0.0001 1 + 0.0110 x0 + -0.0107 x0x0 + 0.0024 x0\_t + 0.0233 x0\_tt + 1.4976 x0x0x0\_t + -0.0414 x0x0\_tt + 1.4973 x0x0x0\_tt  x0x0x0\_t = -0.0002 x0 + 0.0085 x0x0 + 0.0043 x0\_t + -0.0001 x0\_tt + 0.0139 x0x0\_t + 0.0390 x0x0\_tt + 0.0012 x0x0x0\_tt  x0x0\_tt = -0.0001 1 + 0.0073 x0 + -0.2432 x0x0 + 0.0028 x0\_t + 0.0100 x0\_tt + -0.0121 x0x0\_t + 1.3426 x0x0x0\_t + 0.1785 x0x0x0\_tt  x0x0x0\_tt = -0.0027 x0 + 0.0004 x0x0 + 0.0038 x0\_tt + 0.0145 x0x0\_t + 0.0033 x0x0x0\_t + 0.0069 x0x0\_tt |  |

# Looking at train and test data

need to rethink how im including the data

* Instead of using 0hp for train and 1hp for test, splitting 0hp into 2 (one for train, one for test)
* The x\_train data is just 1 column, whereas when using the synthetic data you get 2 columns, and (for ensembling at least) it gives you 2 equations x0’ and x1’
* So how to get 2nd column of x (x1) ?

|  |  |  |
| --- | --- | --- |
| **Test** | **Output** | **Comment** |
| 0hp DE normal operation  80/20 train and test  Ensemble-SINDy  Custom Library:  [1, 'x0', 'x0^2', 'sin(t)'] | Ensemble with replacement (V1)  (x0)' = -0.006 x0^2  Ensemble without replacement (V2)  (x0)' = -0.006 x0^2 | I think this is just plotting test on top of train (with train refitted to match test)  Work out whats actually being plotted here **– it was just test and train being plotted**  Is it time? Or just steps that have been solved |
| 0hp DE normal operation  SINDy-PI | 1 = 1.0000 1  sin(t) = nan 1 + nan sin(t) + nan x0\_t + nan x0\_tt + nan 1 + nan x0 + nan x0^2 + nan x0^3 + nan x0^4 + nan x0^5 + nan x0^6  x0\_t = 0.0003 1 + 0.0013 x0\_tt + 0.0003 1 + 0.0042 x0 + -0.1345 x0^2 + -0.1342 x0^3 + 2.0500 x0^4  x0\_tt = 0.0015 1 + 0.0004 x0\_t + 0.0015 1 + -0.2820 x0 + 0.0203 x0^2 + 1.2337 x0^3 + 2.2796 x0^4  1 = 1.0000 1  x0 = 0.0013 1 + -0.0001 sin(t) + 0.0035 x0\_t + -0.6569 x0\_tt + 0.0013 1 + -0.0439 x0 + 1.3729 x0^2 + 79.2998 x0^3 + -126.8868 x0^4 + -1358.2498 x0^5 + 2225.0030 x0^6  x0^2 = 0.0000  x0^3 = 0.0004 x0\_tt + 0.0087 x0 + 0.0086 x0^2 + -0.0835 x0^4 + 18.0146 x0^5  x0^4 = 0.0001 x0\_tt + -0.0005 x0 + 0.0269 x0^2 + 0.0323 x0^3  x0^5 = -0.0003 x0 + 0.0407 x0^3  x0^6 = 0.0009 x0^2 | ? |

# Ensemble & -pi with normal and fault

* Consider the sample rate for the fault data (12k or 48k)
* It is accelerometer data
* I think its just been plotting x\_test as ‘model’ and x\_train as ‘true’ ???

|  |  |  |
| --- | --- | --- |
| **Test** | **Output** | **Comment** |
| Ensemble SINDy  Plotting the x\_sim term | Ensemble with replacement (V1)  (x)' = 0.002 x + -0.054 x^2    Ensemble model without replacement (V2)  (x)' = 0.002 x + -0.054 x^2 | Simulated over a shorter time to show more detail  Note in Jennifer B.’s logbook for this data, the y axis is ‘amplitude’  Getting error –  UserWarning: Ensembling arguments are deprecated.Use the EnsembleOptimizer class instead.  Check github to see if ensembling working |
| (original) SINDy  Library:  ['1', 'x0', 'x0^2', 'x0^3', 'x0^4', 'x0^5'] | normal  (x0)' = 0.004 x0 + -0.064 x0^2 + -0.132 x0^3 + 0.518 x0^4  *fault*  *(x0)' = -0.001 x0 + 0.008 x0^2 + -0.002 x0^3 + -0.012 x0^4 + 0.006 x0^5*  *combined*  *(x0)' = 0.005 x0 + -0.061 x0^2 + -0.292 x0^3 + 0.386 x0^4 + 3.731 x0^5* |  |

|  |  |  |
| --- | --- | --- |
| **Fault data – 7 mil, 0hp, inner race, DE fault** | | |
| Ensemble SINDy  w/ Library:  [1, 'x0', 'x0^2', 'x0^3', 'sin(t)'] | Training Data  (x0)' = 0.003 x0 + -0.003 x0^2 + -0.174 x0^3    Ensemble with replacement (V1)  (x)' = -0.006 x^2 | I think ‘Training Data’ is recovering with original SINDy ?   * This changes when I repeat it |
|  | Ensemble model without replacement (V2)  (x)' = -0.006 x^2 |  |
| SINDy (original)  Library:  ['1', 'x0', 'x0^2', 'x0^3', 'x0^4', 'x0^5'] | SINDy Model  (x0)' = -0.001 x0 + 0.008 x0^2 + -0.002 x0^3 + -0.012 x0^4 + 0.006 x0^5 |  |
| SINDy-PI  Library:  ['1', 'sin(t)', 'x0\_t', 'x0\_tt', '1', 'x0', 'x0^2', 'x0^3', 'x0^4', 'x0^5', 'x0^6'] | x0\_t = -0.0002 1 + -0.0009 sin(t) + -0.0002 1 + -0.0006 x0 + 0.0114 x0^2 + -0.0025 x0^3 + -0.0159 x0^4 + 0.0063 x0^5 + 0.0009 x0^6    x0\_tt = 0.0133 1 + 0.0133 1 + -1.9958 x0 + -0.0021 x0^2 + -0.0156 x0^3 + 0.0240 x0^4 + 0.0202 x0^5 + -0.0093 x0^6 | The SINDy-PI model it very different depending on which feature is plotted   * This means is not finding a sparse solution b/c in theory the only equations in the model should all be the same (if rearranged) |
| Ensemble-SINDy  Opt threshold = 0  Library:  [1, 'x0', 'x0^2', 'x0^3', 'sin(t)'] | Ensemble with replacement (V1)  (x)' = -0.001 x^2    Ensemble model without replacement (V2)  (x)' = -0.001 x^2 | I checked that I was def refitting model and rewriting x\_sim for V2  But still the exact same |
|  | Library:  [1, 'x0', 'x0^3', 'sin(t)']  Ensemble with replacement (V1)  (x)' = -0.001 x^2  Ensemble model without replacement (V2)  (x)' = -0.001 x^2 | So im not actually using the library ive defined for the model?  Turns out ive been re-calling model w/out including library as an argument – so its only been using the default library – polynomial library (from pysindy.py in utils) |

# Changing the library

* I think I can only include sin(t) if using SINDy-PI

|  |  |  |
| --- | --- | --- |
| Test | Output | Comment |
|  |  |  |

# Combining normal + faulty

* From Jennifer B.

|  |  |  |
| --- | --- | --- |
| **Method** | Appending faulty data to normal operation data | |
| **Test** | **Output** | **Comment** |
| SINDy  w/ Library:  ['1', 'x0', 'x0^2', 'x0^3', 'x0^4', 'x0^5'] | SINDy Model  (x0)' = 0.005 x0 + -0.061 x0^2 + -0.292 x0^3 + 0.386 x0^4 + 3.731 x0^5 | (in normal sindy file)  No noticeable change at any point |
|  |  |  |
|  |  |  |