Classification of proto-neutron star oscillation modes

Guillem Fernández-Rodríguez
Phd Student in Mathematics at University of Valencia

VNIVERSITAT DE VALÈNCIA

Collaborators:
Isabel Cordero-Carrión (University of Valencia)
Dimitra Tseneklidou (University of Valencia)

Introduction

In CCSNe, we are interested in the PNS oscillation modes. We can classify them based on a:

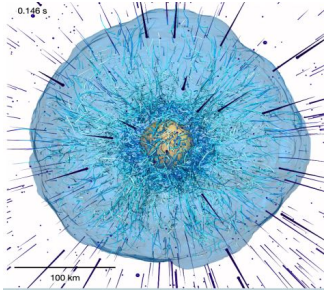
- Physically motivated way, based on the energy of the restoring forces of the modes.

simulations

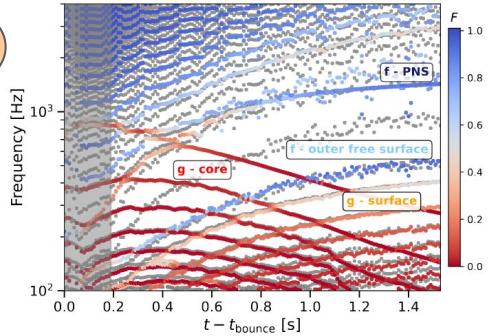
Previous Dimitra's talk

- Data-driven way, by using similarity based criteria between discovered modes.

observations



Burrows et al (2021)



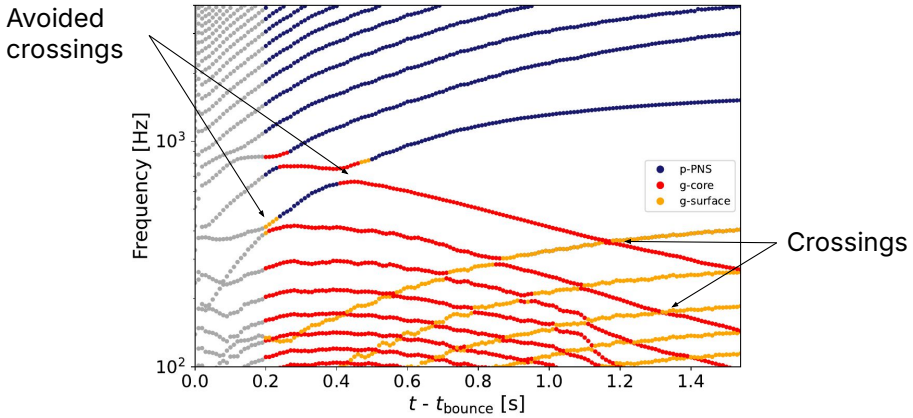
D. Tseneklidou et al (2026)

arXiv:2608.10264

Introduction

Assumptions

1. Modes of the same family are (linearly) similar
2. There is a separation between modes of the same family
3. ? At certain small x axis intervals, the modes behave linearly (*)



Simulation Data

2D Simulation S20 LS220
from M. Obergaulinger

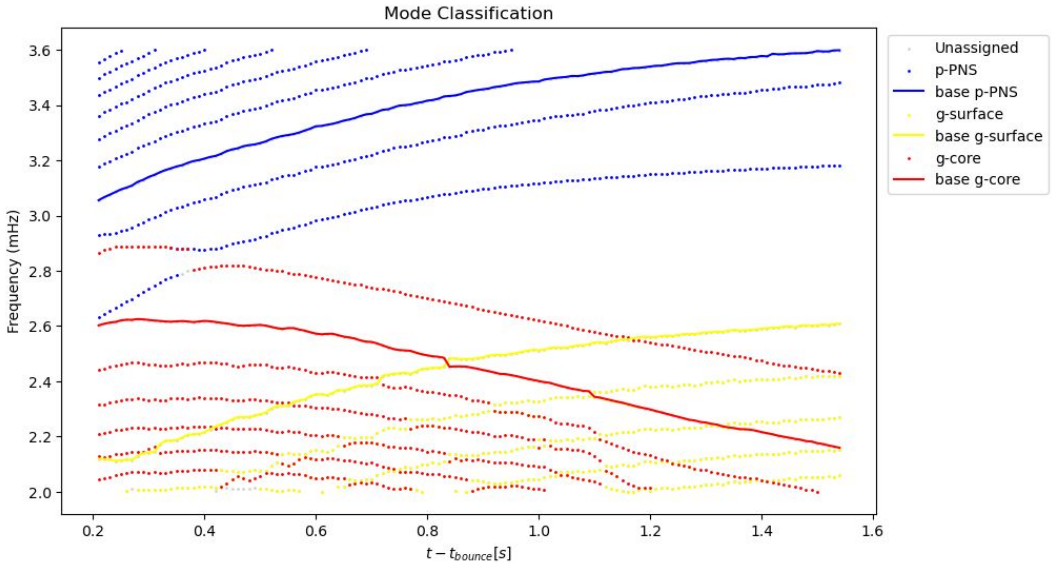
Perturbative analysis done until surface

D. Tseneklidou et al (2026)
arXiv:2608.10264

Introduction

Objectives

- ☑️ Mode classification method given a single member from each family.
- ☐ Classification given no known base modes.



Last CoConut meeting: We manually gave the continuous lines as base modes of each family to the algorithm and classified the others.

The two key points

1 Use **correlation** between modes

2 Prolong based on **separation distance**

The two key points and the one who survived

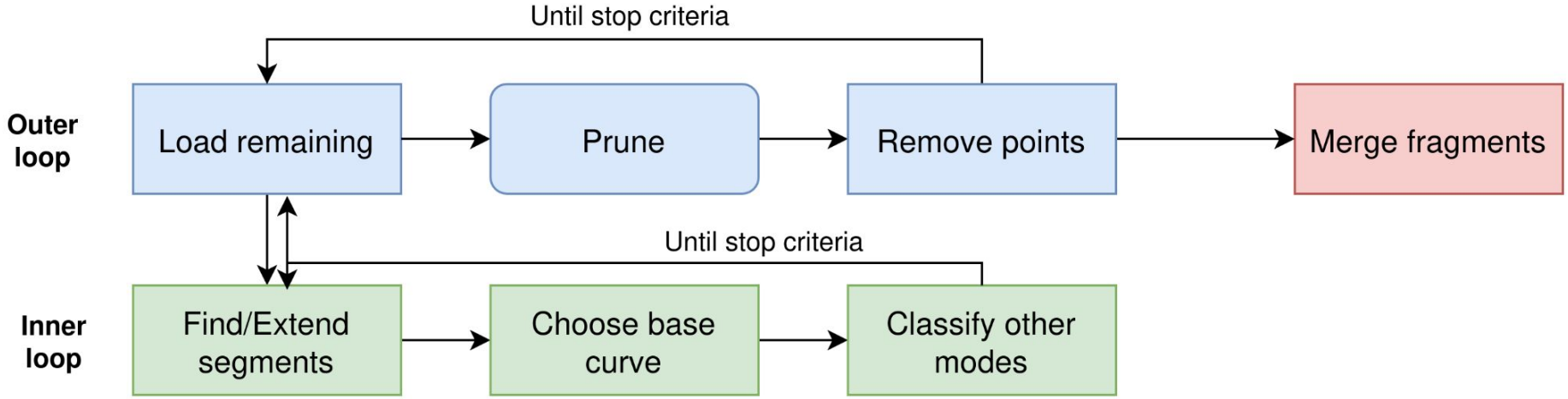
1 Use **correlation** between modes

2 Prolong based on **separation distance**

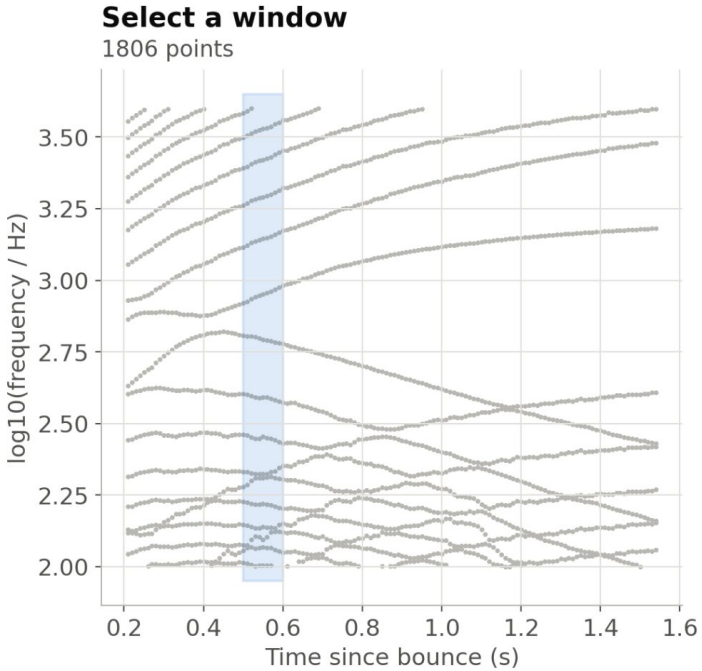
? Find lines in short segments

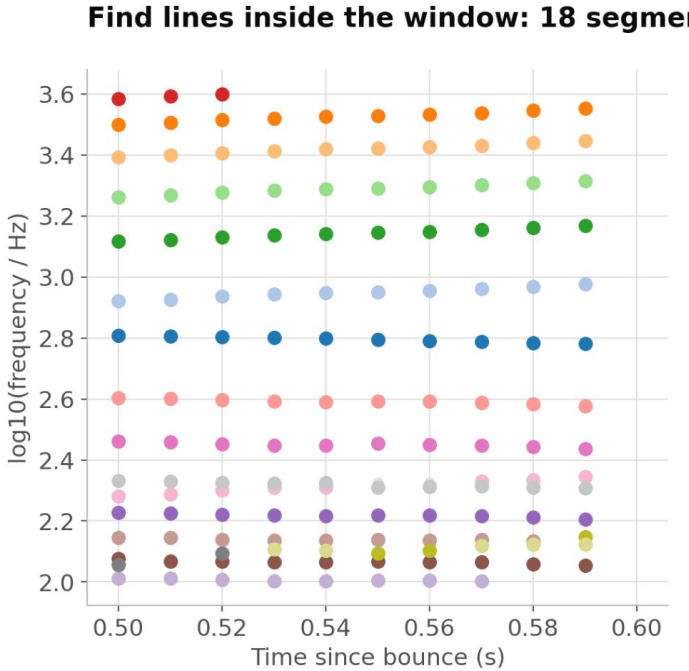
And the technical story is...

Algorithm


Until stop criteria
Outer loop
Load remaining
Prune
Remove points
Merge fragments
Until stop criteria
Inner loop
Find/Extend segments
Choose base curve
Classify other modes

Inner loop: find initial segments

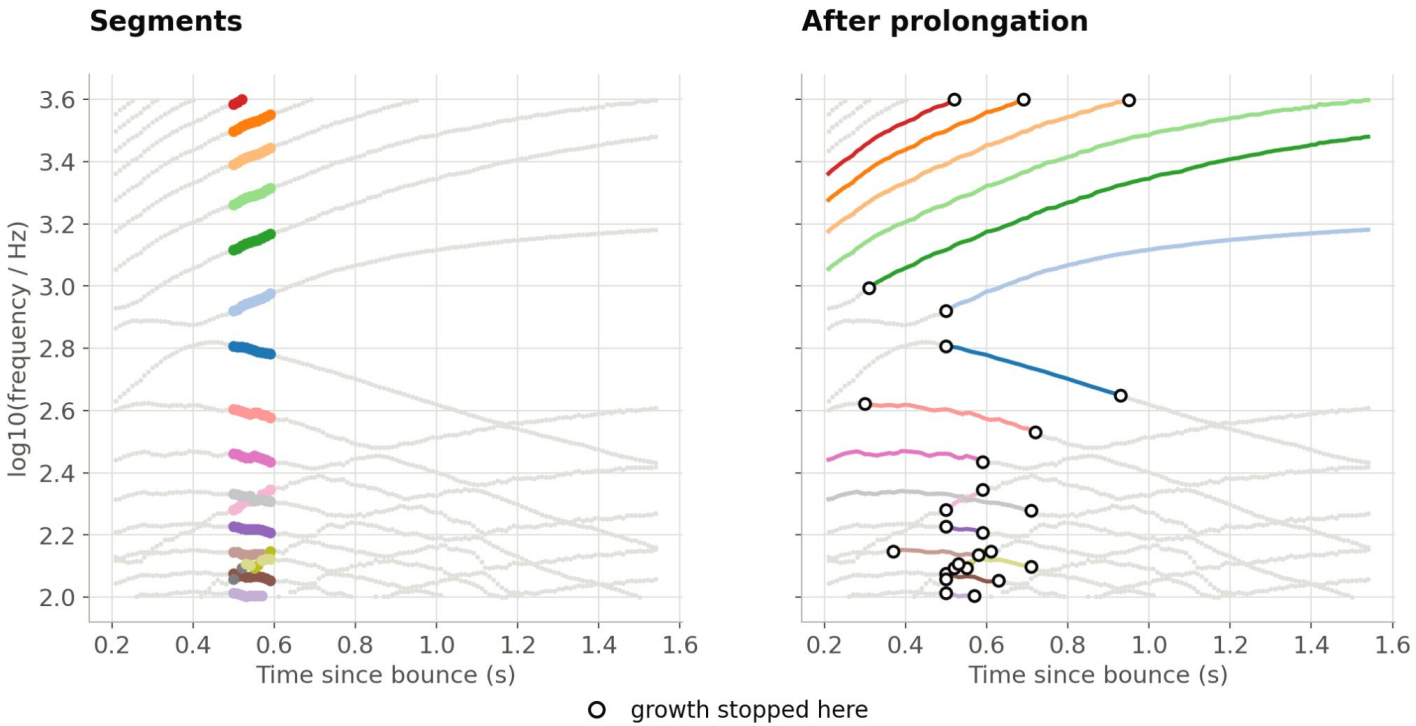




Find/Extend segments

1. Find densest segment
2. Look for local lines with j-linkage

Inner loop: extend segments



Find/Extend segments

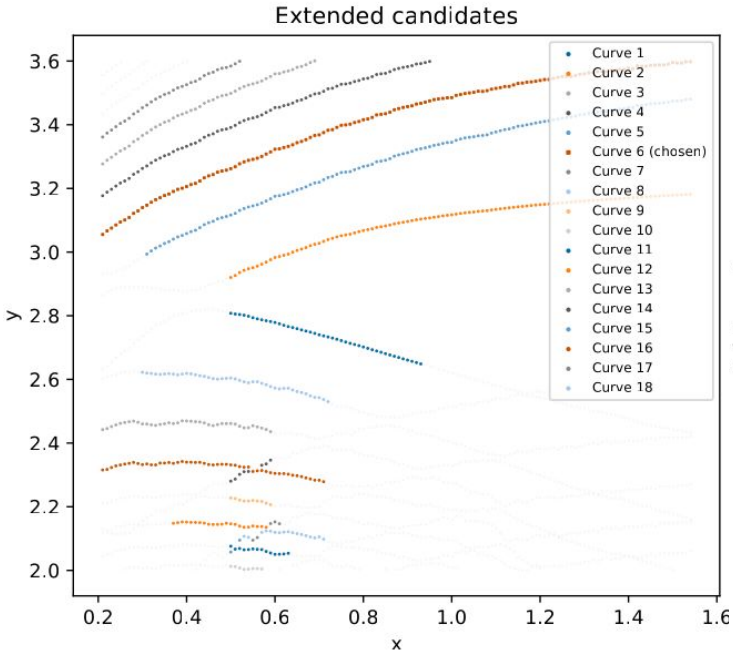
Prolongation algorithm

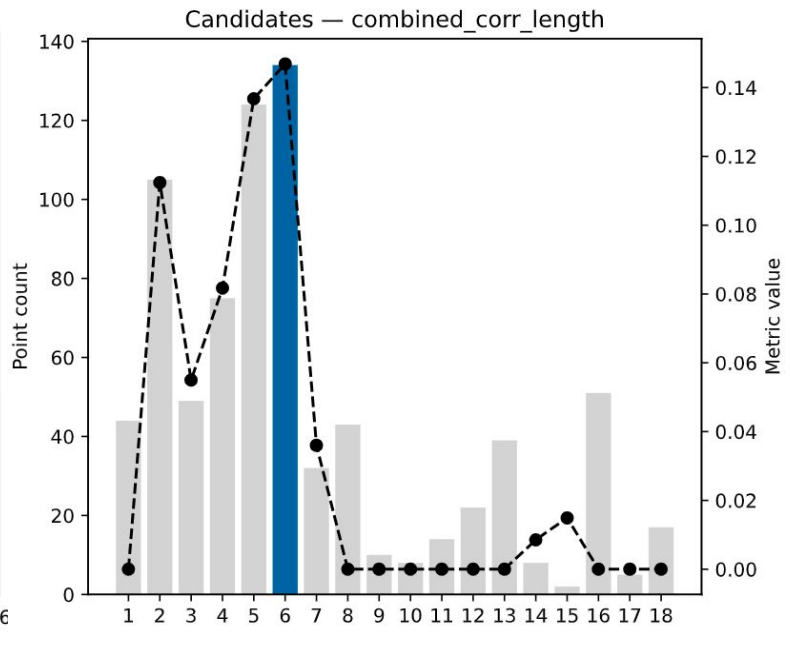
Extend while:

$$d_j < w_i d_i$$

d_i : distance to higher and lower points

Inner loop: choose base curve



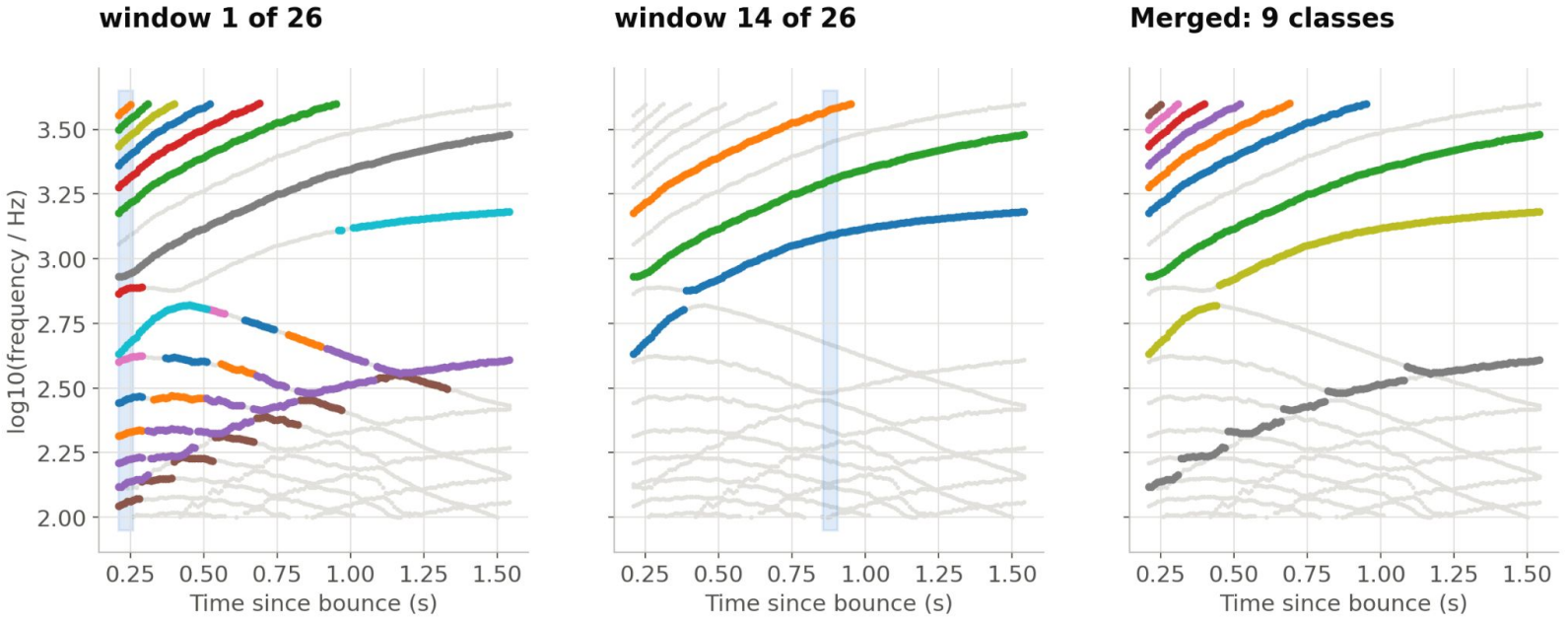


Choose base curve

Score is combination of:

- Length
- Correlations

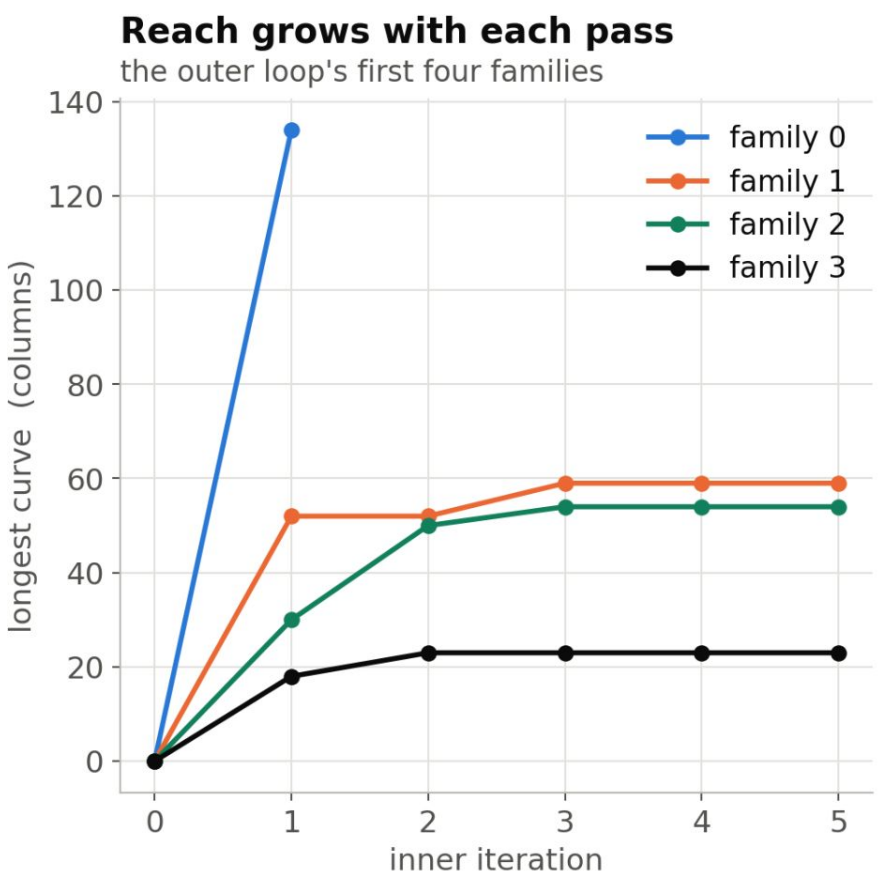
Inner loop: classify other modes



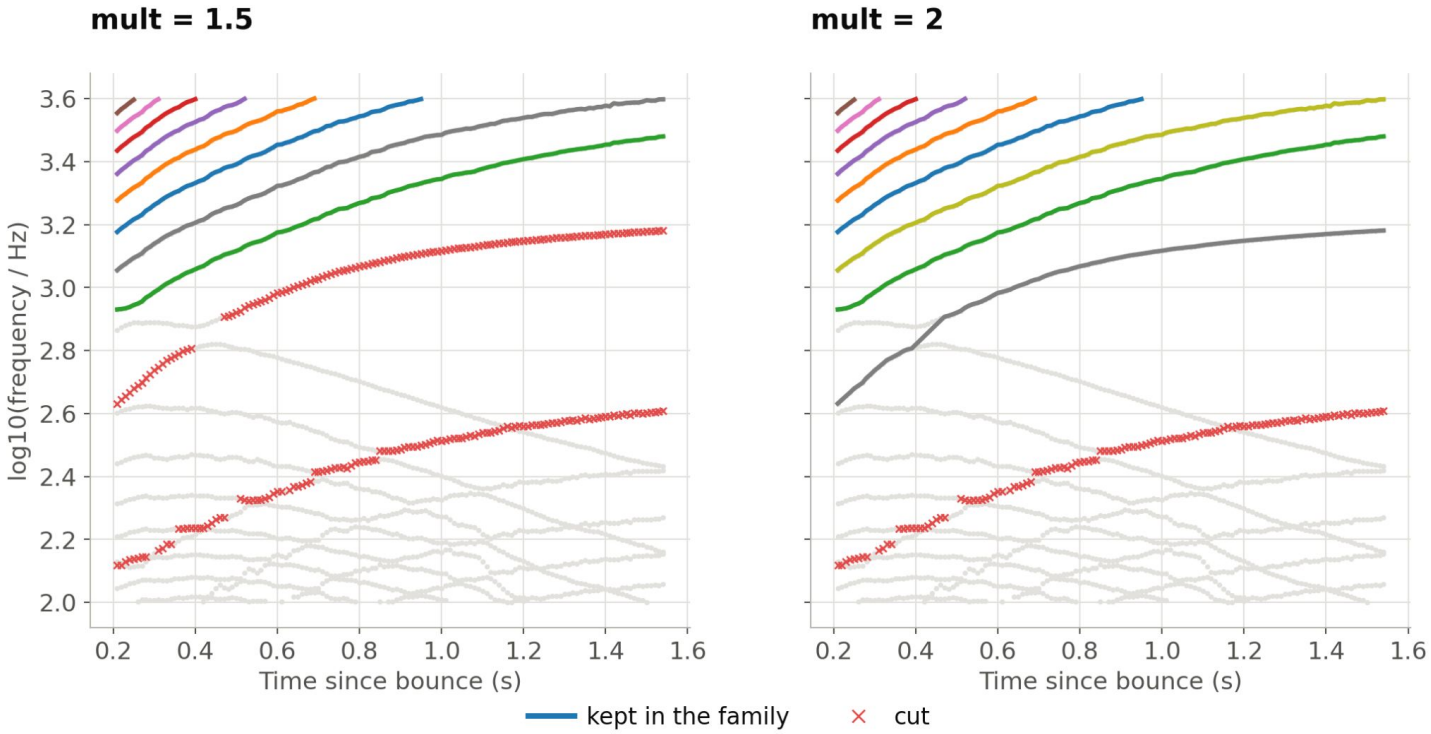
Classify other modes

Importance of inner loop

In the inner loop:
⟶ We classify modes using the base curve and use the previous classification to find a better base curve.
⟶ Consider at least 3-4 iterations.



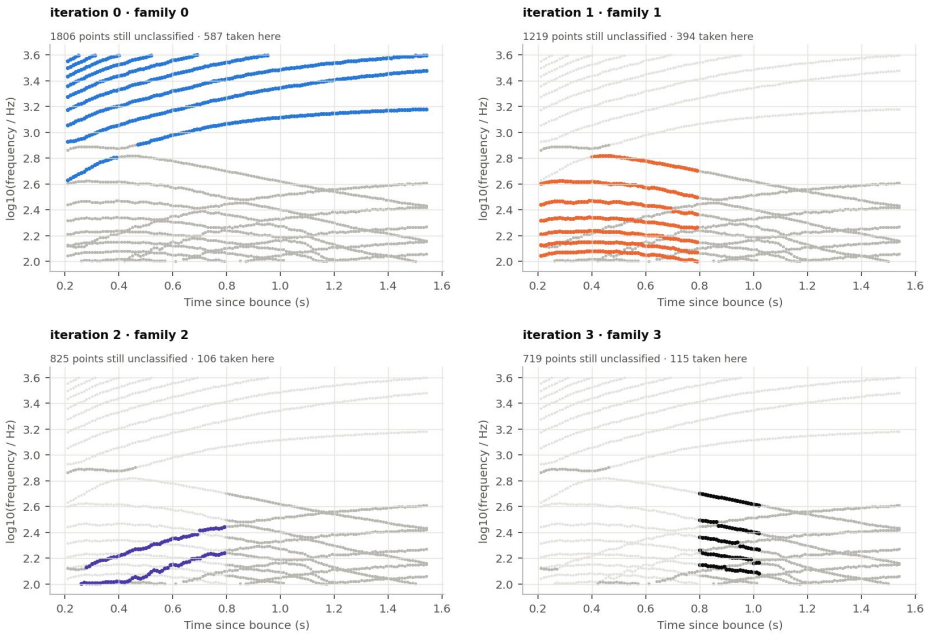
Outer loop: prune

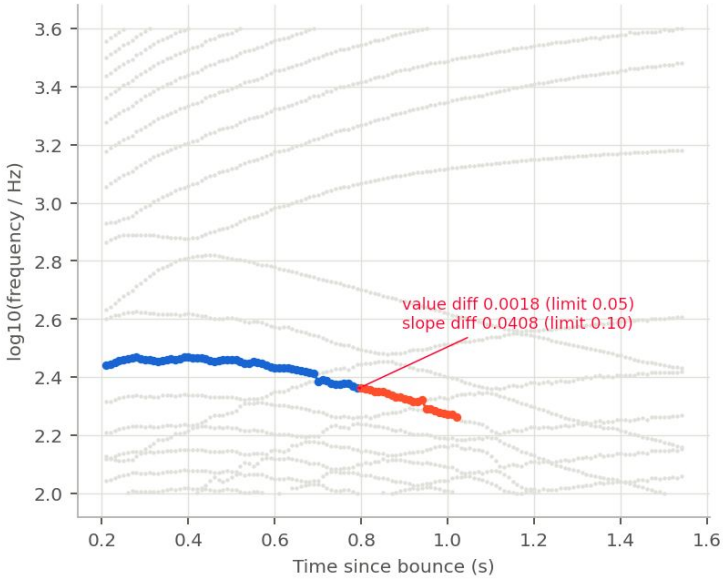


Prune

→after consecutive median distance passes a threshold

Post loops: Merging families

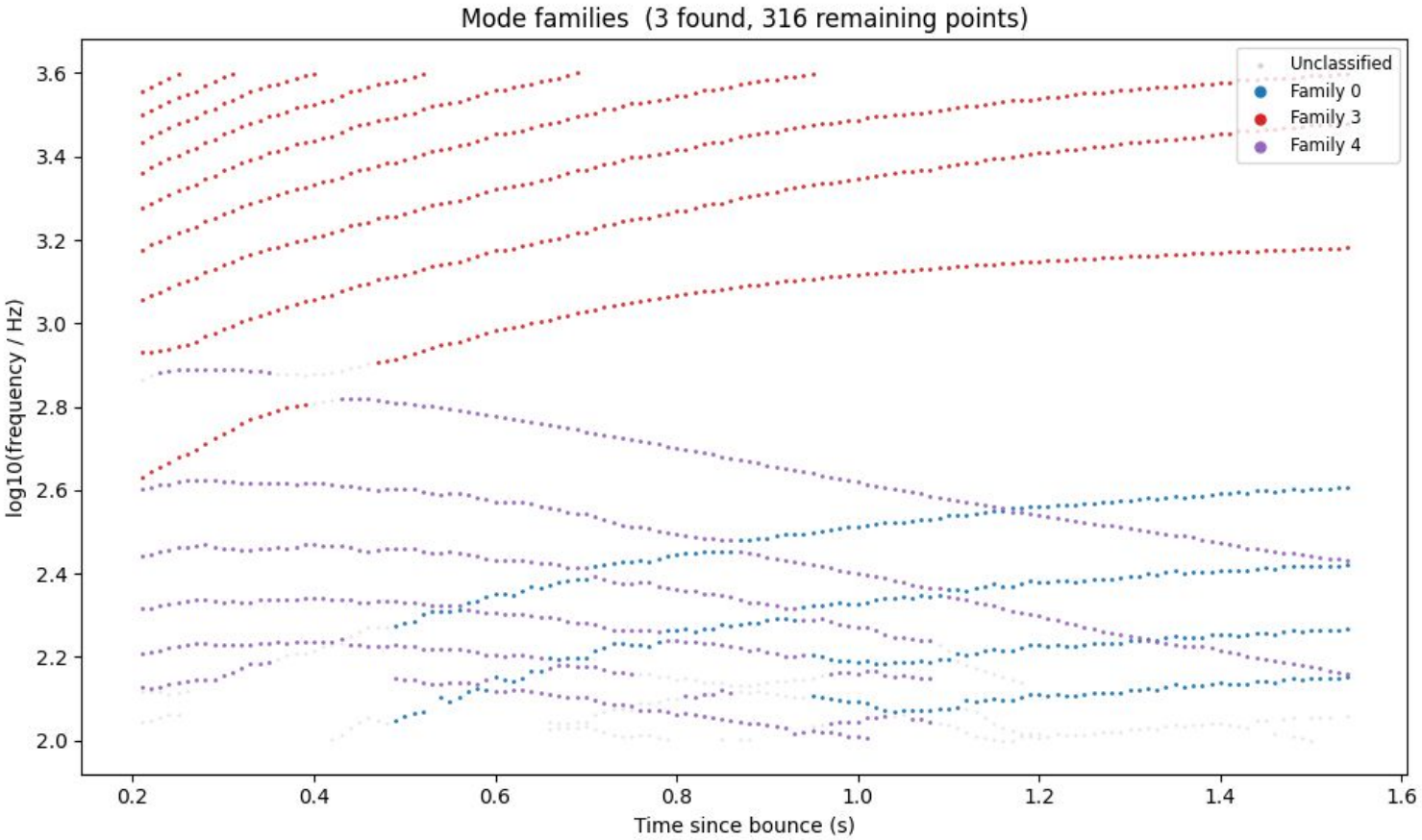




Merge fragments

After outer iterations finish, we try to merge them based on the slope at the connection.

Results



Mixed modes near the boundary are very difficult to classify.

→ Scoring method
→ Separation factor

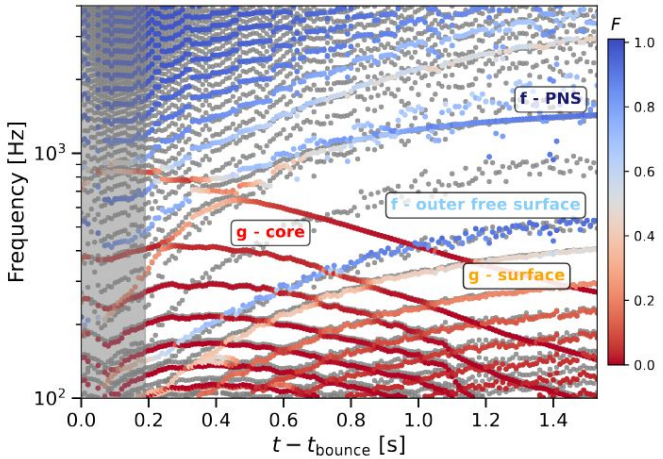
Application to other datasets

Oscillatory dataset (🏗️ Work in progress)

with boundary conditions at the shock, and 4 families

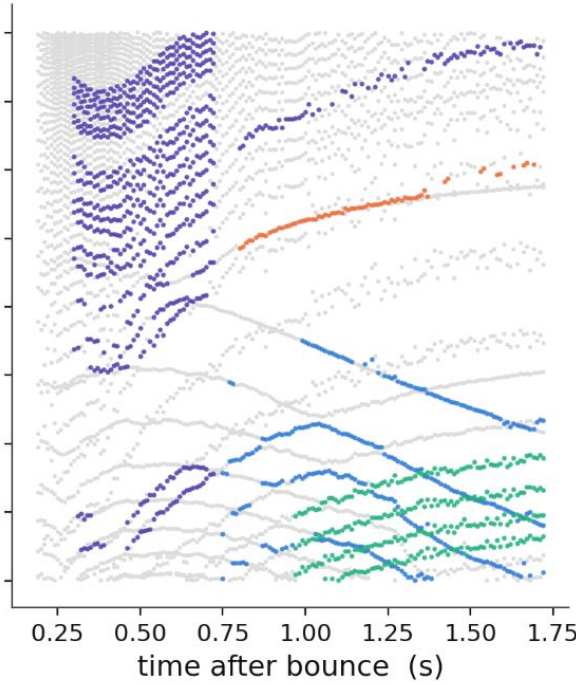
⟶ Separation between modes is much smaller

⟶ Last assumption is not true



≠

Results



Conclusions

✅Pros

- Data-driven method useful for observation only applications
- Fast *low-tech* algorithm
- Fully automatic except some parameter tuning
- Robust method to crossings and avoided crossings

❌Cons

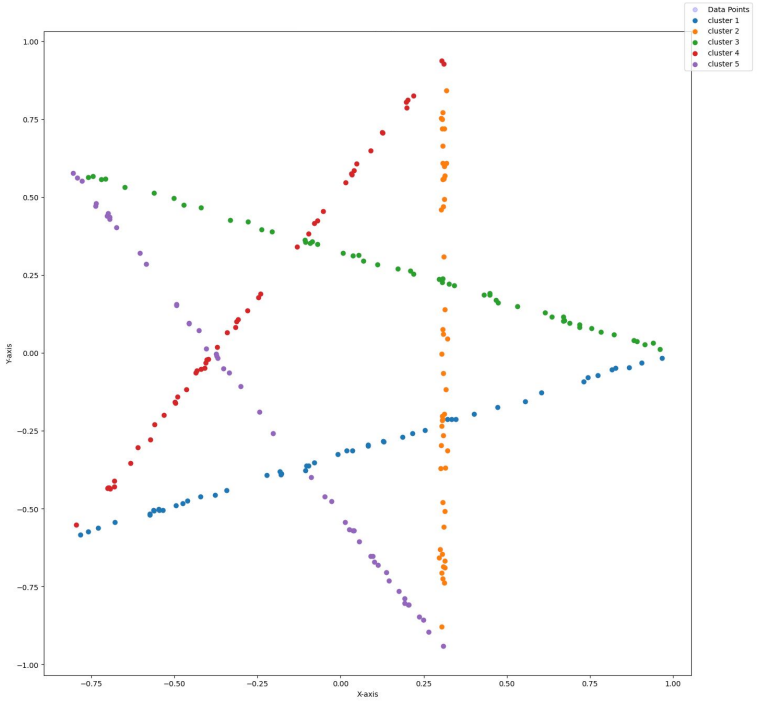
- Classification is limited near noisy boundaries
- Not robust to non-smooth oscillation modes

🔮 Future work

- Method robust to oscillatory modes
- Application to other datasets

THANK YOU

Extra: Line classification with J-Linkage



Toldo, R. (2008)
DOI:10.1007/978-3-540-88682-2_41

- Minimal Sample Sets

$$P(\mathbf{x}_j|\mathbf{x}_i) = \begin{cases} \frac{1}{Z}\exp -\frac{||\mathbf{x}_j-\mathbf{x}_i||^2}{\sigma^2} & \text{if } \mathbf{x}_j \neq \mathbf{x}_i \\ 0 & \text{if } \mathbf{x}_j = \mathbf{x}_i \end{cases}$$

- Each point is represented by which models it belongs to.
- Model ownership is determined by distance to it.
- Clustering using Jaccard distance

$$d_J(A,B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}.$$