ETH zürich

Data Science Lab, Spring 2024

Seismic Fault Reconstruction Algorithms

Frederieke Lohmann, Sam Houliston
Supervised by Dr. Men-Andrin Meier

Secondary supervisors: Daniil Dmitriev, Prof. Fanny Yang

Background

Goal: Accurate Reconstruction from Point Clouds

- Quakes are induced by sudden displacements of earth along faults.
- Sensors can capture such displacements as point-clouds, which can be clustered to model the fault.
- Improved modelling of faults is key to understand their propagation directions, the risk of subsequent quakes, and general properties of surrounding material.

Particularities of Quake Data

- Faults are planar and usually aligned in one main direction
- Fractal nature, sparse and noisy $\implies$ difficult to cluster!

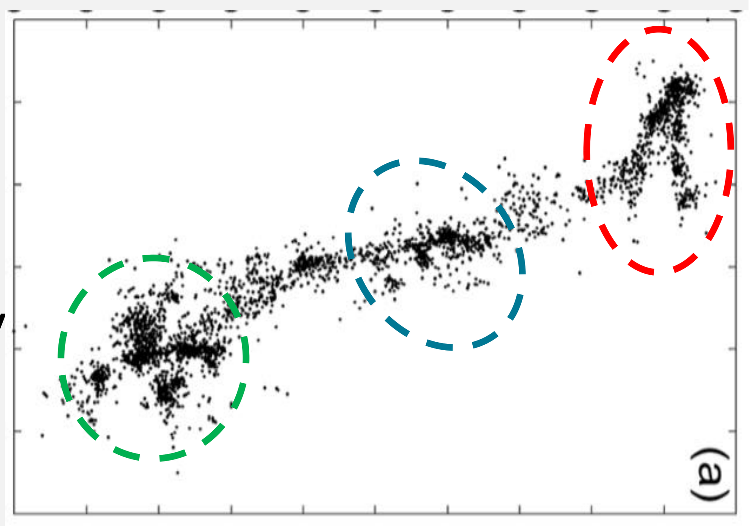


Figure 1: Landers dataset of real measurements

Problem:

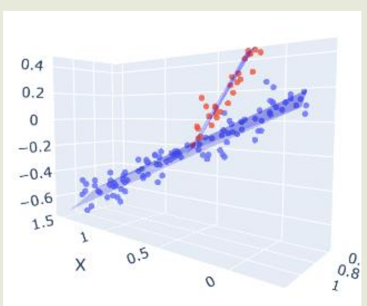
- Standard algorithms perform poorly on seismic data
- Few tailored algorithms exist and fail on test cases
- Real datasets are expensive to label
  $\implies$ no quantitative evaluation to develop new algorithms!

Contributions:

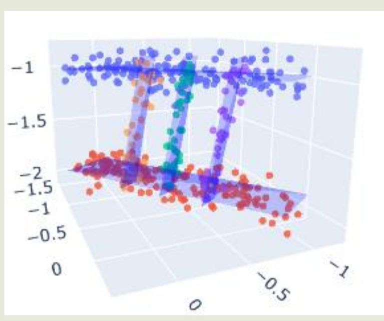
1. **Developed a realistic synthetic data generator**
2. **Implemented a cutting-edge algorithm in python**
3. **Improved this probabilistic clustering algorithm**

Synthetic Data Generation

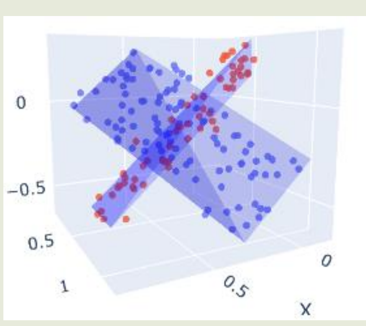
We generate synthetic data based off expert knowledge by simulating key fault structures via planes from which point clouds are sampled.



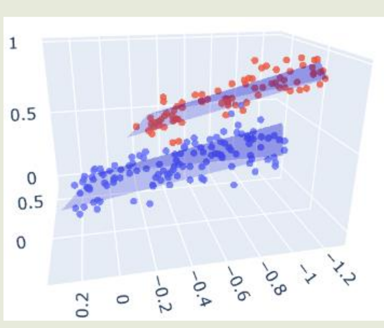
Y-shaped



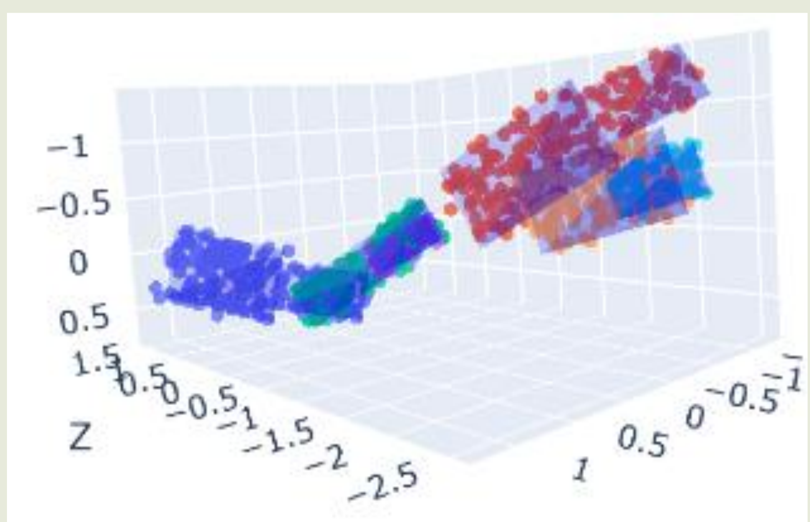
ladder



cross



parallel



Example dataset

Figure 2: Example Synthetic Faults and Mixed Dataset

- Basic fault plane geometry taken from rigorous models
- We identify 6 common fault patterns: straight, bent, parallel, Y-shaped, cross faults and a ladder-like structures.
- A dataset comprises of realizations of fault cases, generated randomly along one of two main 'preferred axes'

The Bedretto Dataset is a high-resolution dataset of 2723 points, it is highly planar and features one ladder. Measurements were collected over 48hrs from a deliberately induced quake. We perform experiments on first 15hrs of data.

Our Proposed Clustering Algorithm

We base our method on a cutting edge probabilistic agglomerative method, which fits gaussian kernels to clusters and merges coinciding kernels if statistically relevant.

We introduce a margin penalty to favour cluster planarity, and an additional re-fitting step of the mixture.

Algorithm 1 Agglomerative Clustering by Kamer et al.

1: **Input:** Point-cloud dataset of measurements.
2: **Output:** Gaussian kernels that represent individual fault clusters.

3: **I. Initialize Capacity Clusters and Gaussian Kernels**
- Form agglomeration tree of clusters based on Ward's criterion
- Pick level with most clusters of size bigger than `min_cluster_size`
- Initialize mixture model:
  1. Fit gaussian kernel to valid 'capacity' clusters ($\geq$ `min_cluster_size`)
  2. Fit uniform background kernel to remaining points
- Fit one bounding box per cluster/kernel

4: **II. Agglomerative Probabilistic Clustering**
While unconverged (Gain > 0):
  For every pair of clusters with overlapping bounding boxes:
  - Compute merged kernel (sum gaussian)
  - Compute Gain from merging:
    $\mathrm{Gain_{BIC}} = \mathrm{BIC_{mixture\_before}} - \mathrm{BIC_{mixture\_after\_merging}}$
    $\mathrm{Gain} = \mathrm{Gain_{BIC}} - \lambda\,(1 - \mathrm{alignment\_score(pair)}) > 0$

  While pairs with Gain > 0 remaining:
  - Merge pair (A,B) with highest Gain
  - Remove all other pairs containing A or B
  - Remove pairs (C,D) having positive gain with D or C (ex: Gain(A,C)>0)
- Re-assign points to kernels with maximum likelihood
- Re-fit mixture model of gaussian + background kernels.

5: **Return:** Gaussian kernels representing fault segments

- Gaussians are representative of fault distributions.
- We implemented the method in python, originally in MATLAB.
- The margin penalty favours merging aligned clusters.
- The alignment score is the dot-product between the normals of two clusters.
- Lambda value of same order of magnitude as $\mathrm{Gain_{BIC}}$.
- Re-fitting mixture model at every iteration ensures higher accuracy.

Evaluation

Results on Synthetic Data

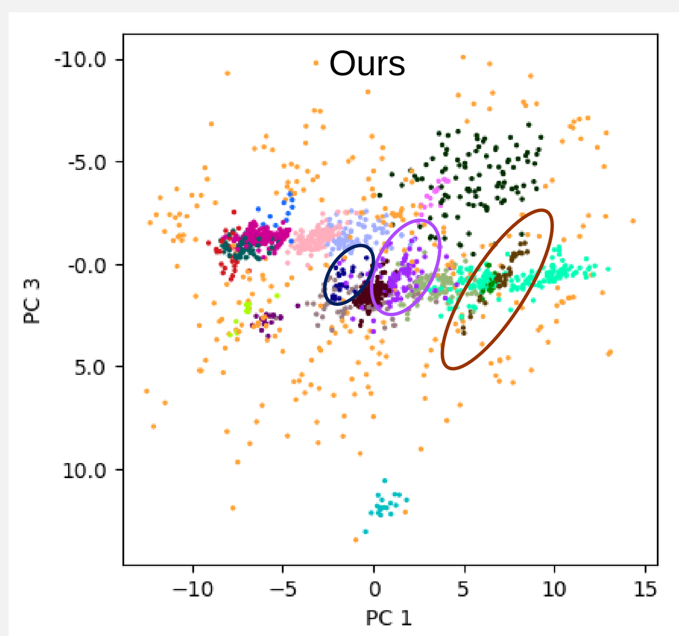
- Our method performs better than the baselines on both tricky individual structures and larger datasets.
- Adjusted Rand index measures similarity between 2 clusterings.

Performance Comparison

	Ours	Kamer et al.	K-Means
Y-fault	**0.723**	0.616	0.076
Cross	**0.775**	0.673	0.365
Ladder	**0.863**	0.736	0.051
Mixed Dataset	**0.810**	0.728	0.495

Table 1: Mean Adjusted Rand index w.r.t. ground truth over 20 trials. Mixed Dataset comprises of 15 faults.

Result on the Bedretto Experiment



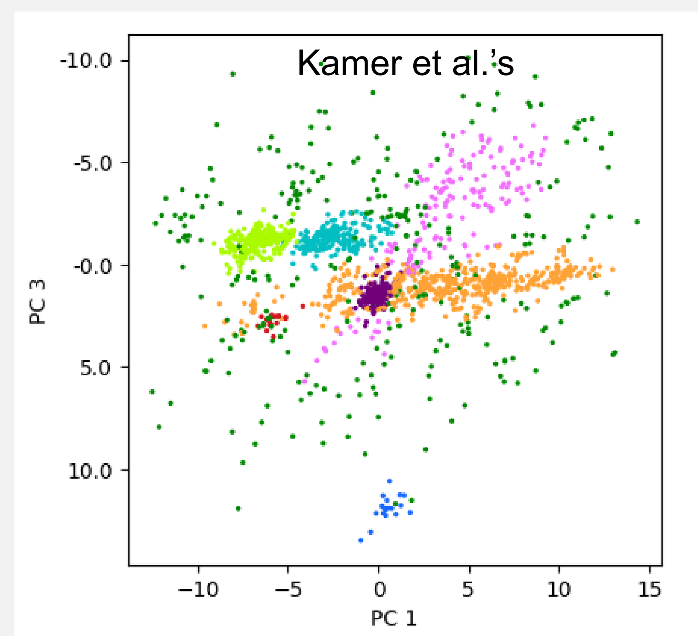


Figure 3: Clusterings on the Bedretto dataset. We visualize the data along principal components PC1 and PC3. Hyperparameters: Ours: lambda=16, min_cluster_size=10; 15 for Kamer et al.. Our method better clusters the 'rungs' of the ladder structure, giving a finer clustering.

Discussion

- Our method shows improvements quantitatively on synthetic data and gives a good clustering on real noisy datasets.
- The penalty strength is tunable, ideal value may vary between datasets.
- Future work could make the algorithm more efficient by merging phases I and II, or partitioning data in a principled manner for parallel computation.
- Could integrate plane-favouring methods ex RANSAC (not probabilistic).