# Laplacian Score for Feature Selection

Xiafei He, Deng Cai, Partha Niyogi, 2005

Jérémie Dentan and Gonzague de Carpentier
Professors: Laurent Oudre and Charles Truong

March 2023

# Outline

# Outline

# Introduction: Feature selection

Paper under review: *Laplacian Score for Feature Selection* [5].

- Why do we want to select features? [4]
  - Better predictive performances
  - Computational efficiency
  - Need to measure fewer features
  - Interpretability
- What kind of method exist for that?
  - **Wrapper methods**: Feature selection wrapped around task learning.
  - **Filter methods**: Feature selection prior to task.
    - Supervised: use labels
    - Unsupervised: without labels

The **Laplacian Score** is an unsupervised filter method.
Idea: Preserve the structure of the nearest neighbors graph.

# Outline

# Laplacian Score

1. **Compute the nearest neighbor graph** $G$**:**

$$G_{i,j} := \begin{cases} 1 & \text{if } x_i \text{ is among the } k \text{ nearest neighbors of } x_j \text{ or reciprocally} \\ 0 & \text{otherwise} \end{cases}$$

2. **Compute the weighted adjacency matrix** $S$**:**

$$S := G \odot \exp\left(-\frac{1}{\sigma^2} M^2\right) \in \mathbb{R}^{m \times m}$$

3. **Compute the degree matrix** $D$**:** $D := \operatorname{diag}(S\mathbb{1}) \in \mathbb{R}^{m \times m}$
4. **Compute the centered features** $\tilde{f}$**:** $\tilde{f}_r = f_r - \frac{f_r^T D \mathbb{1}}{\mathbb{1}^T D \mathbb{1}} \mathbb{1}$
5. **Compute the laplacian scores** $L_r$**:**

$$L_r := \frac{\tilde{f}_r^T L \tilde{f}_r}{\tilde{f}_r^T D \tilde{f}_r} \in [0, 1] \qquad L := D - S$$

6. **Select the features** having the highest Laplacian scores.

# Our experiments

What will we do?

- Evaluate the impact of the **hyperparameters** $\sigma$ and $k$
- Evaluate the impact of using **DTW** or the euclidian distance
- **Compare** the method to classical feature selection methods: (1) a simple variance threshold (unsupervized) and (2) filtering on the ANOVA score [7] (supersized).

How do we measure the performance?

- `sklearn`'s SVC with default parameters
- Measure the accuracy for binary classification based on the same set of features

# Outline

# Data

Three datasets from https://timeseriesclassification.com:

- **Earthquakes** [1]:
  - Data: readings from Northern California Earthquake Data Center
  - Labels: major earthquake event or not
- **Wafer** [6]:
  - Data: process control measurements during the processing of silicon wafers
  - Labels: normal or abnormal
- **WormsTwoClass** [3]:
  - Data: projection of the motion of worms on a particular dimension, second-long intervals
  - Labels: wild-type or mutant

# Data visualization

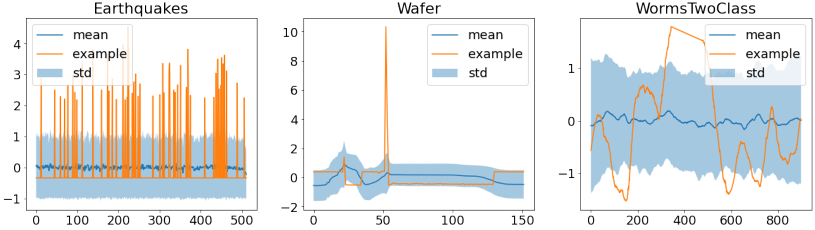


Figure: Visualization of the three datasets. For each dataset, we plot the average time series, the standard deviation at each timestamp and an example sampled randomly from the dataset.

# Autocovariance functions

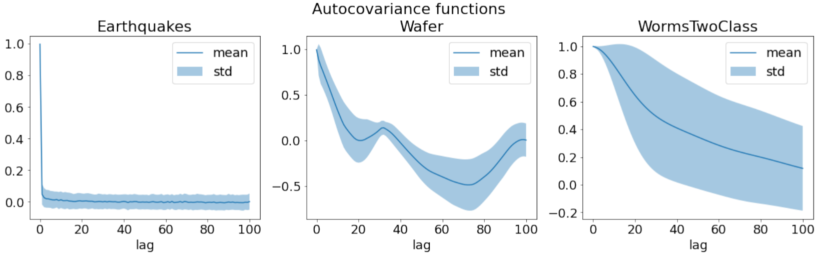


Figure: Average autocovariance functions for the three datasets.

Then, we used TSFEL [2] to extract the features.

# Outline

# Distribution of the Laplacian Scores

3 regimes:

- $\sigma$ small: $S \to 0$, scores concentrated around 0
- Transition phase
- $\sigma$ huge: $S \to G$, so the scores also converge

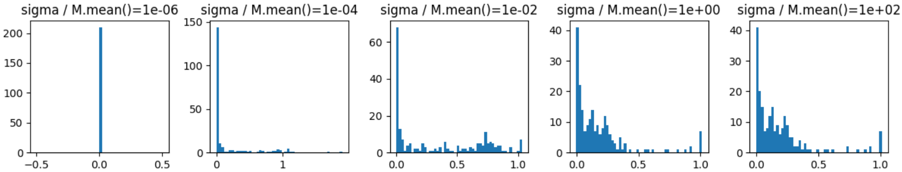


Figure: Histograms of the value of the Laplacian score for several values of $\sigma/\overline{M}$

# Influence of $\sigma$

- For some datasets, the task is either too simple or too difficult
- Good heuristic: take $\sigma$ quite small

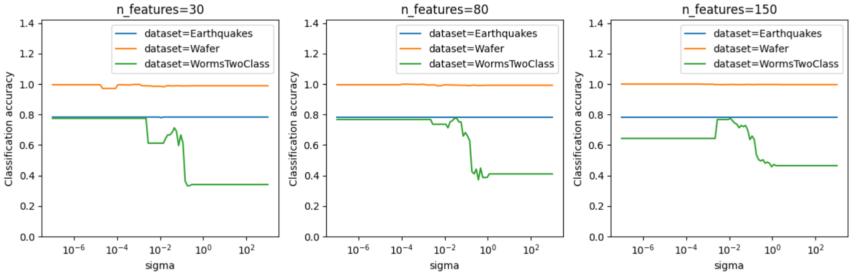


Figure: Evolution of the classification accuracy against the value of $\sigma$.

# Influence of the number of nearest neighbors

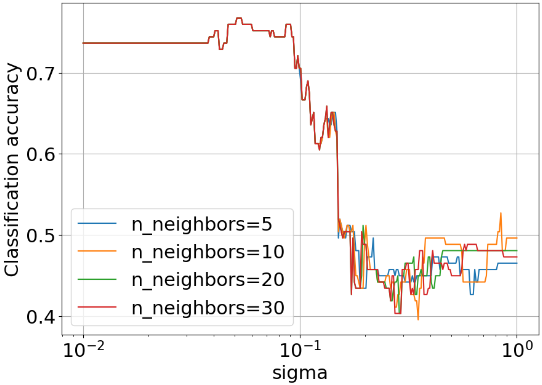


Figure: Evolution of the classification accuracy against the value of sigma.

$\Rightarrow$ Good heuristic: take $\sigma/\overline{M}$ small, around $10^{-4}$, and $k$ medium, of the order of ten.

# Comparison with other selection methods

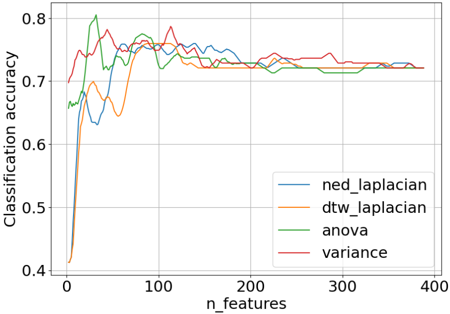


Figure: Evolution of the classification accuracy against the number of features.

- Similar, but slightly lower performance
- DTW is not better than euclidian distances here

*NED = Normalized Euclidian Distance*

# Conclusion

- Advantage: unsupervised method
- Drawback: 2 hyperparameters to tune. Not very stable.
- Interesting method but perfomance on tested datasets and task is not overwhelming.

# References

[1] Anthony Bagnall. *Earthquakes dataset*.

[2] Marília Barandas et al. "TSFEL: Time Series Feature Extraction Library". Jan. 2020.

[3] Andre Brown and Anthony Bagnall. *WormTwoClass dataset*.

[4] Isabelle Guyon and André Elisseeff. "An introduction to variable and feature selection". Mar. 2003.

[5] Xiaofei He et al. "Laplacian Score for Feature Selection". 2005.

[6] Robert Thomas Olszewski. *Wafer dataset*.

[7] Henry Scheffé. *The Analysis of Variance*. Mar. 1999.