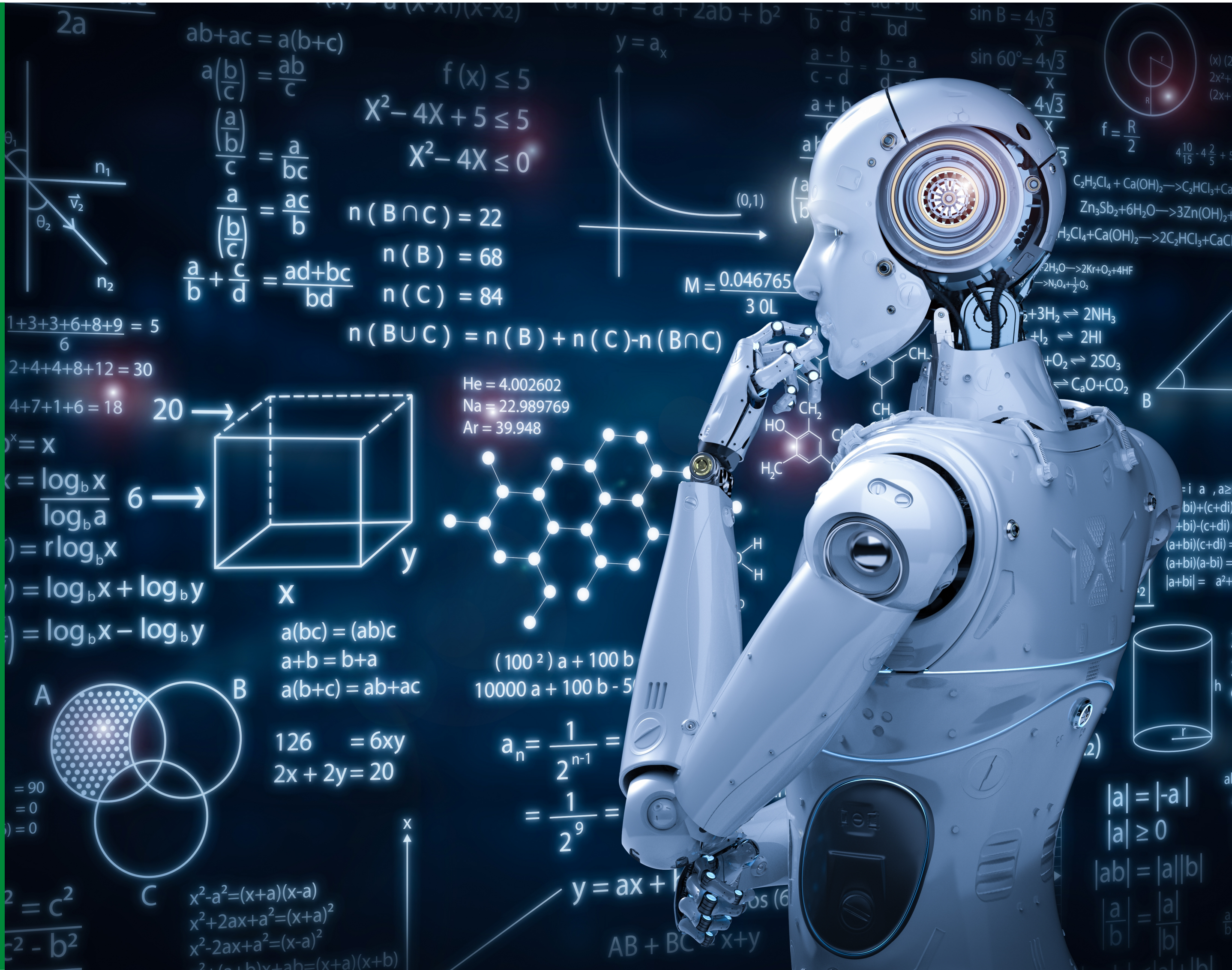
A Fractional Graph Laplacian Approach to Oversmoothing

S. Maskey*, R. Paolino*,
A. Bacho, G. Kutyniok

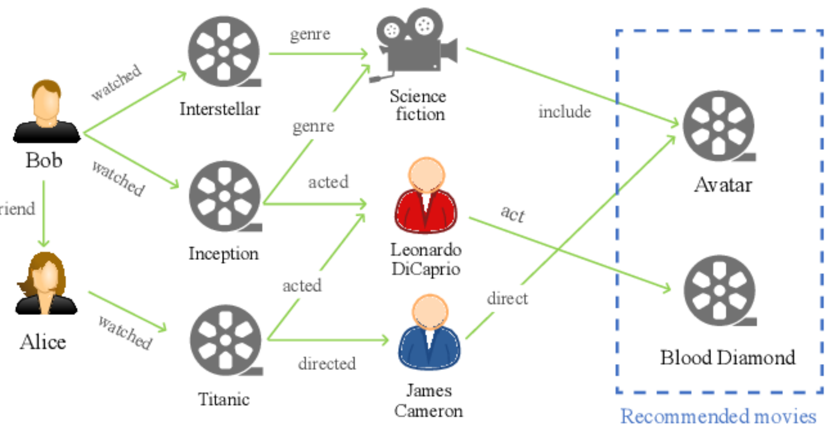
Department of Mathematics - LMU Munich



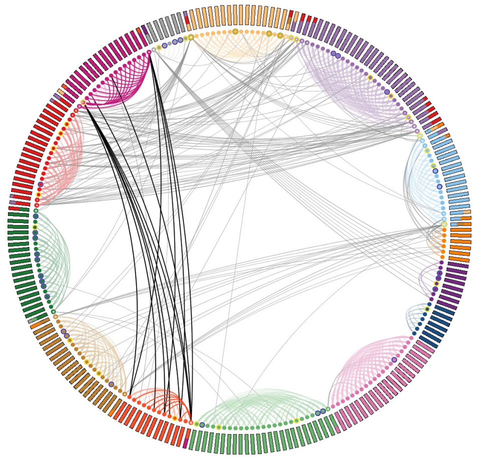
The Current Impact of Graph Neural Networks

Social Networks

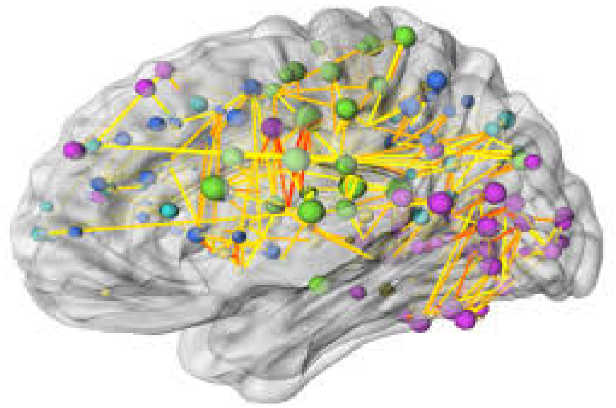
Recommender Systems



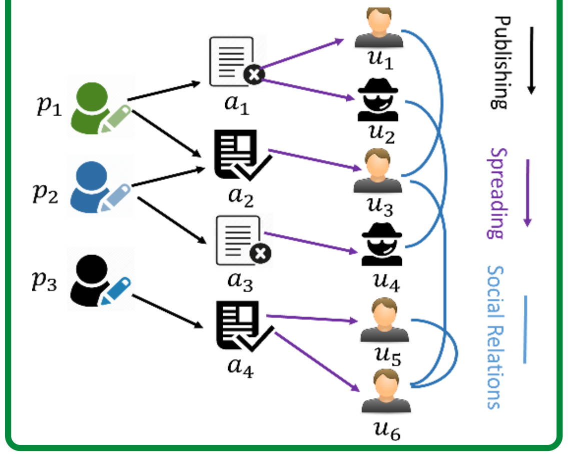
Drug Discovery



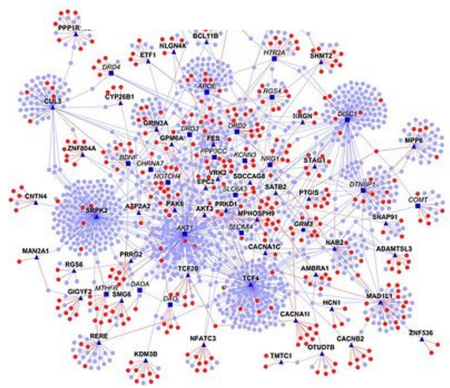
Functional Brain Networks



Fake News Detection

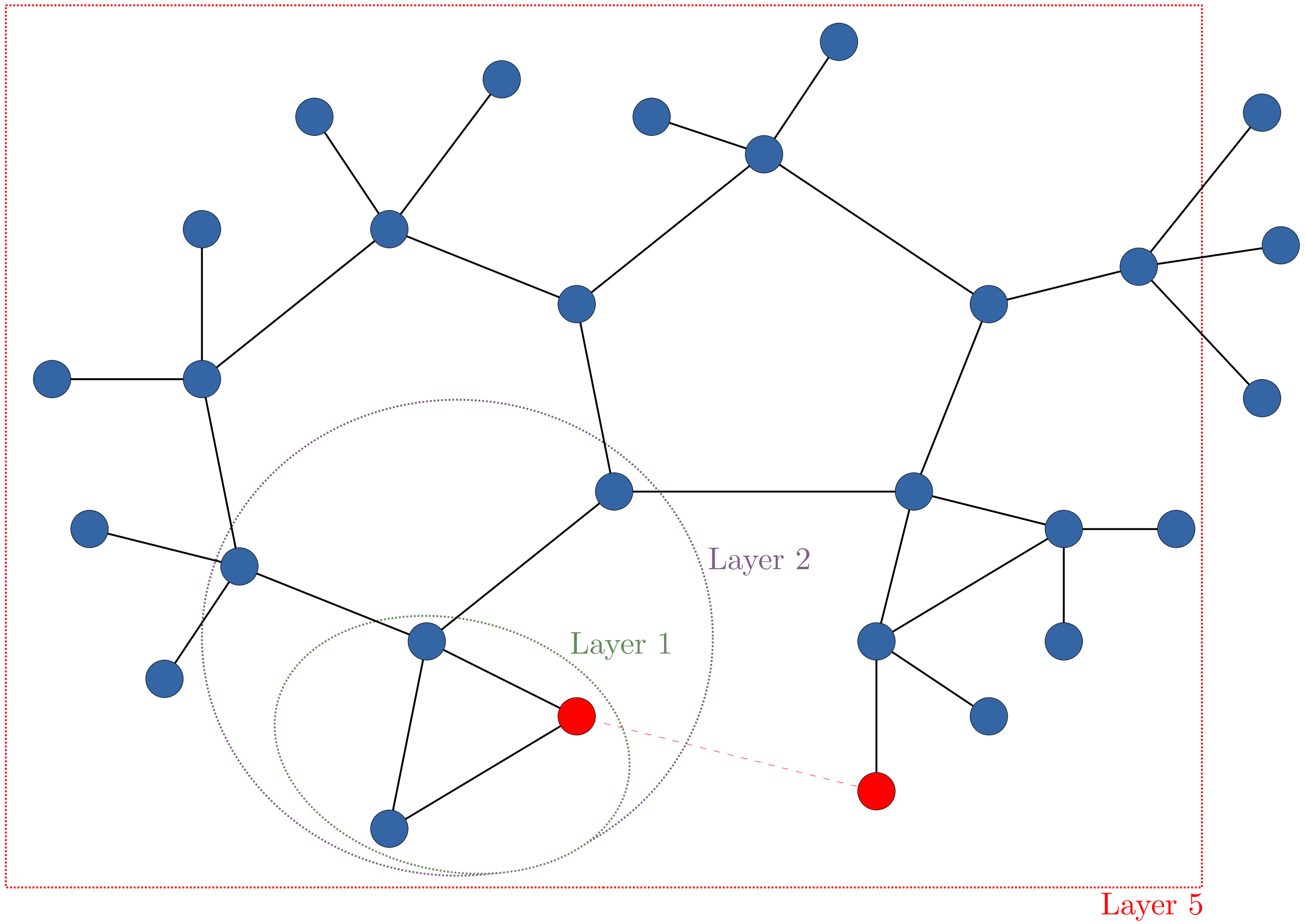


Drug Design



Gilmer, J., et al. (2017). *Neural message passing for quantum chemistry.*
Wang, J., et al. (2018). *Billion-scale commodity embedding for e-commerce recommendation in alibaba.*
Monti, F., et al. (2019). *Fake news detection on social media using geometric deep learning.*
Fan, W., et al. (2019). *Graph neural networks for social recommendation.*

GNNs Fail to Capture Long-Range Interactions



→ To capture long-range dependencies, GNNs need increased depth.

→ The receptive field increases exponentially fast[1,2].

- → All nodes have the same computational graph.
- → All nodes get the same embedding.
- → The nodes' features converge to similar values.

▶Oversmoothing

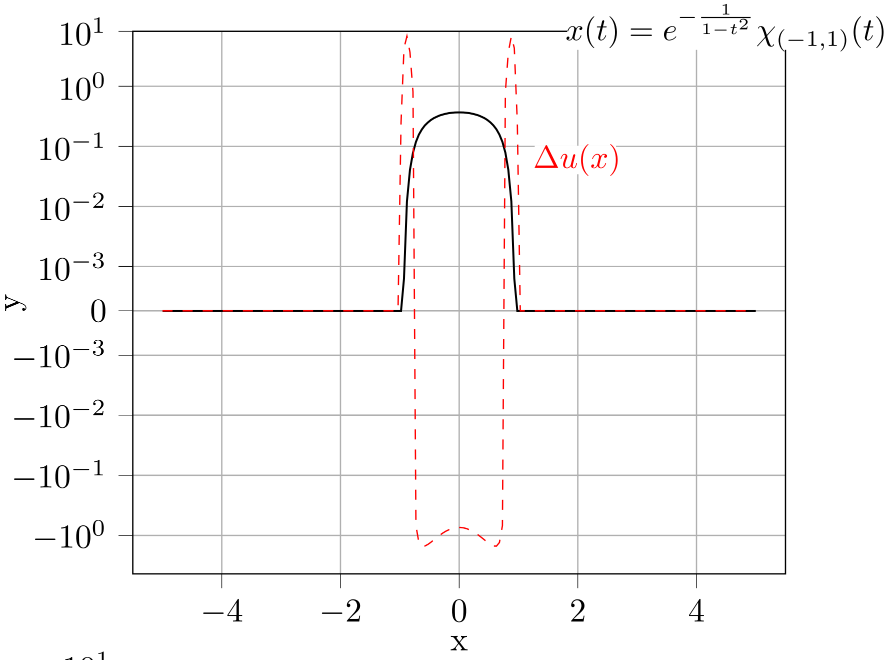
→ Not analyzed in directed graphs.

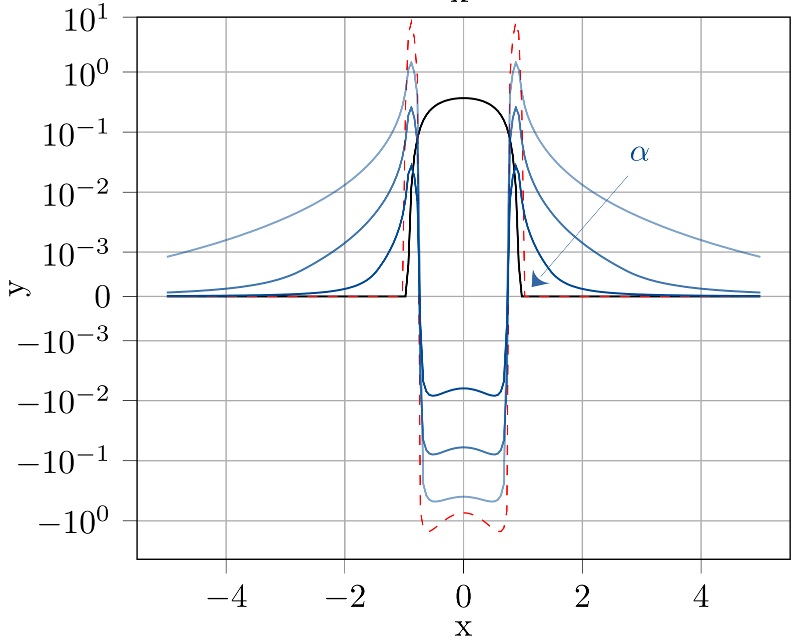
Can we take inspiration from physics to address it?

[1] Oono, K., Suzuki, T. (2019). *Graph Neural Networks Exponentially Lose Expressive Power for Node Classification.*

[2] Cai, C., Wang, Y. (2020). *A Note on Over- Smoothing for Graph Neural Networks.*

Non-Local Diffusion





GNNs propagate node features similarly to the heat equation

$$\begin{cases} x'(t) = -\Delta\, x(t)\,, \\ x(0) = x_0. \end{cases}$$

where the Laplacian is a differential ($\to$ local) operator

$$-\Delta x(t) = \lim_{r \to 0} c_n \int_{|t|<r} \frac{x(t) - x(s)}{r^{n+2}} ds.$$

Solution: replace the local operator $(-\Delta)$ with the global operator $(-\Delta)^{\alpha}$ defined as

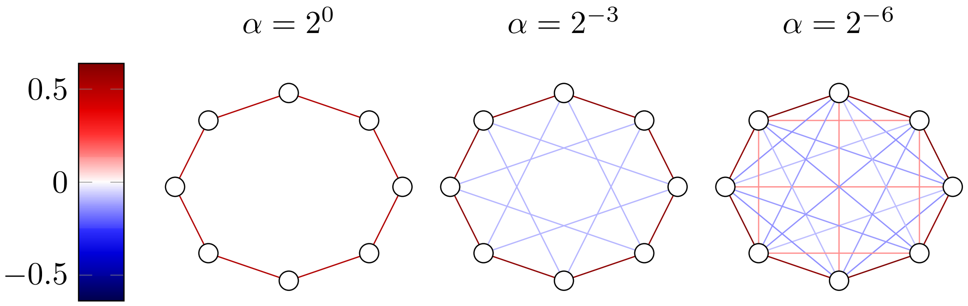
$$(-\Delta)^{\alpha} x(t) = c_{n,\alpha} \int_{\mathbb{R}^n} \frac{x(t) - x(s)}{|t-s|^{n+2\alpha}} ds, \; \alpha \in (0,1).$$

$\to$ The support increases as α decreases.

Fractional Graph Laplacian

Fractional Graph Laplacian

$$\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}^{\mathsf{H}} = \mathrm{SVD}(\mathbf{L}),\ \mathbf{L}^{\alpha} := \mathbf{U}\mathbf{\Sigma}^{\alpha}\mathbf{V}^{\mathsf{H}}.$$



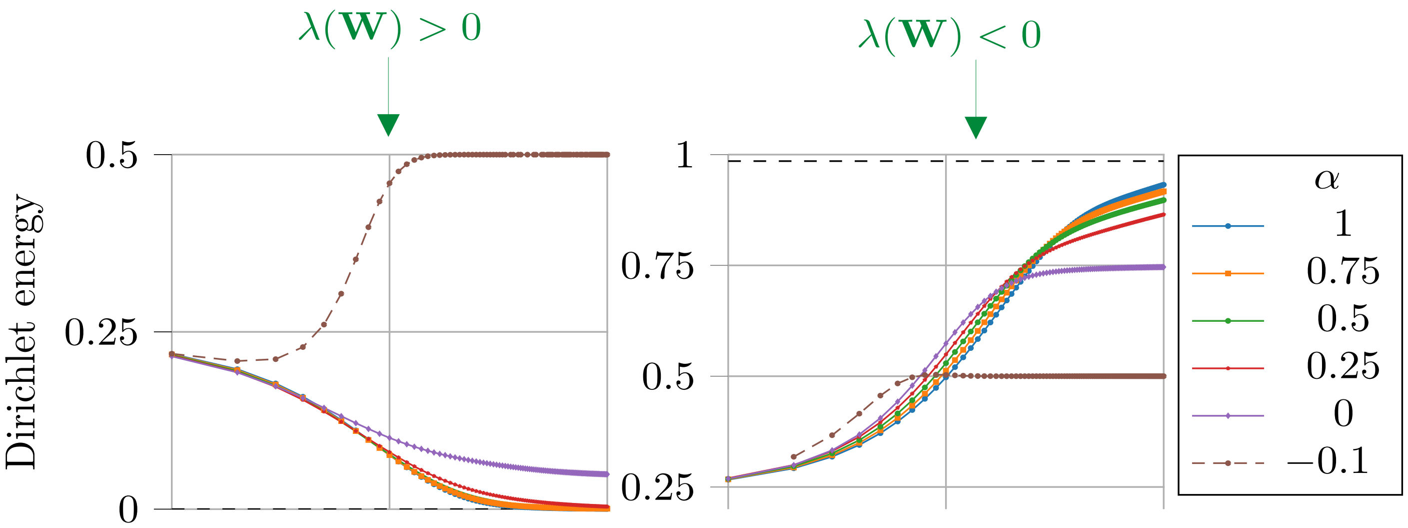
Fractional Graph Heat Eq.

$$\mathbf{x}'(t) = -\mathbf{L}^{\alpha}\,\mathbf{x}(t)\,\mathbf{W},\ \mathbf{x}(0) = \mathbf{x}_0.$$

solution →

$$\mathrm{vec}\,(\mathbf{x})\,(t) = \exp\left(-\left(\mathbf{W} \otimes \mathbf{L}^{\alpha}\right) t\right) \mathrm{vec}\,(\mathbf{x}_0).$$

Cora (undirected) →



```
% A, x_0 are given.
% Preprocessing
D_in  = diag(A1)
D_out = diag(A^T 1)
L = D_in^{-1/2} A D_out^{-1/2}
U, Σ, V^H = svd(L)
% α, h, W learnable parameters
% x_0 initial nodes' features
def training_step(x_0):
    x_0 = input_MLP(x_0)
    % Forward Euler Scheme
    for n ∈ {1, ..., N} do
        x_n = x_{n-1} - i h U Σ^α V^H x_{n-1} W
    x_N = output_MLP(x_N)
    return x_N
```

GitHub

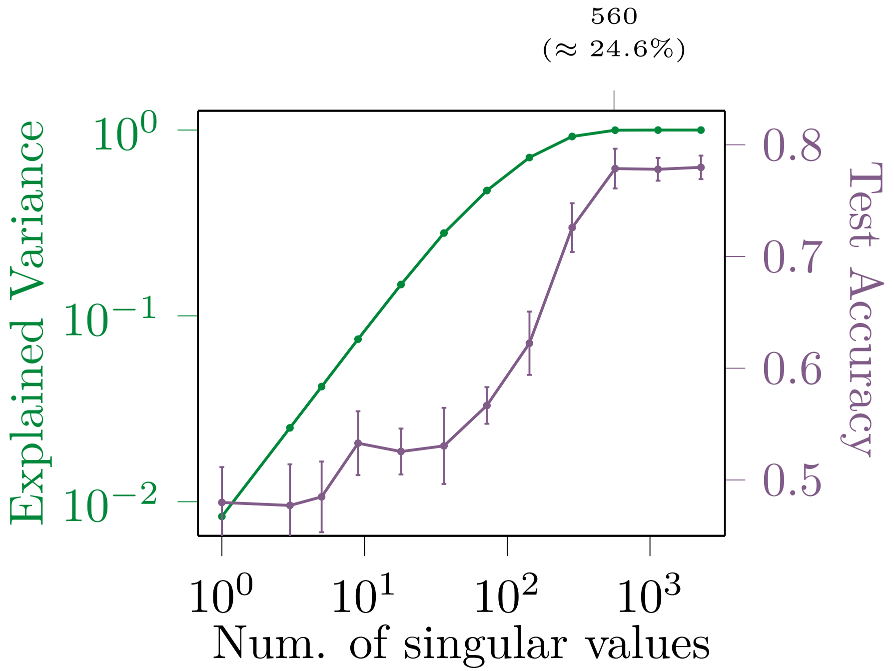
fLode

Disadvantages:

1. Computational cost grows cubically in N.
2. Storage cost grows quadratically in N.

Advantages:

1. Easy to implement.
2. Versatile across different types of graphs.
3. Reduced cost with truncated SVD.



Thank you very much for your attention!

Maskey*, S., Paolino*, R., Bacho, A., Kutyniok, G. (2023). *A Fractional Graph Laplacian Approach to Oversmoothing.*