Permutation Equivariance of Graph Filters

- We will show that graph convolutional filters are equivariant to permutations

Definition (Permutation matrix)

A square matrix $\mathbf{P}$ is a permutation matrix if it has binary entries so that $\mathbf{P} \in \{0, 1\}^{n \times n}$ and it further satisfies $\mathbf{P}\mathbf{1} = \mathbf{1}$ and $\mathbf{P}^T\mathbf{1} = \mathbf{1}$.

- The product $\mathbf{P}^T\mathbf{x}$ reorders the entries of the vector $\mathbf{x}$.
- The product $\mathbf{P}^T\mathbf{S}\mathbf{P}$ is a consistent reordering of the rows and columns of $\mathbf{S}$

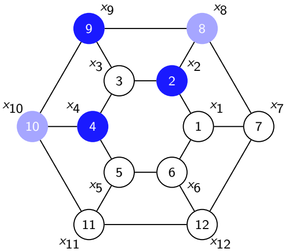
Definition (Permutation matrix)

A square matrix $\mathbf{P}$ is a permutation matrix if it has binary entries so that $\mathbf{P} \in \{0,1\}^{n \times n}$ and it further satisfies $\mathbf{P}\mathbf{1} = \mathbf{1}$ and $\mathbf{P}^T\mathbf{1} = \mathbf{1}$.

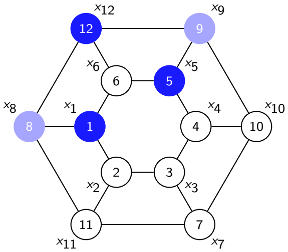
- Since $\mathbf{P}\mathbf{1} = \mathbf{P}^T\mathbf{1} = \mathbf{1}$ with binary entries $\Rightarrow$ Exactly one nonzero entry per row and column of $\mathbf{P}$
- Permutation matrices are unitary $\Rightarrow \mathbf{P}^T\mathbf{P} = \mathbf{I}$. Matrix $\mathbf{P}^T$ undoes the reordering of matrix $\mathbf{P}$

- If $(\mathbf{S}, \mathbf{x})$ is a graph signal, $(\mathbf{P}^T\mathbf{S}\mathbf{P}, \mathbf{P}^T\mathbf{x})$ is a relabeling of $(\mathbf{S}, \mathbf{x})$. Same signal. Different names

Graph signal $\mathbf{x}$ Supported on $\mathbf{S}$



Graph signal $\hat{\mathbf{x}} = \mathbf{P}^T\mathbf{x}$ supported on $\hat{\mathbf{S}} = \mathbf{P}^T\mathbf{S}\mathbf{P}$



- Processing should be label-independent $\Rightarrow$ Permutation equivariance of graph filters and GNNs

- Graph filter $\mathbf{H}(\mathbf{S})$ is a polynomial on shift operator $\mathbf{S}$ with coefficients h_k. Outputs given by

$$\mathbf{H}(\mathbf{S})\mathbf{x} = \sum_{k=0}^{K-1} h_k \mathbf{S}^k \mathbf{x}$$

- We consider running the same filter on $(\mathbf{S}, \mathbf{x})$ and permuted (relabeled) $(\hat{\mathbf{S}}, \hat{\mathbf{x}}) = (\mathbf{P}^T\mathbf{S}\mathbf{P}, \mathbf{P}^T\mathbf{x})$

$$\mathbf{H}(\mathbf{S})\mathbf{x} \;=\; \sum_{k=0}^{K-1} h_k \mathbf{S}^k \mathbf{x} \qquad\qquad \mathbf{H}(\hat{\mathbf{S}})\hat{\mathbf{x}} \;=\; \sum_{k=0}^{K-1} h_k \hat{\mathbf{S}}^k \hat{\mathbf{x}}$$

- Filter $\mathbf{H}(\mathbf{S})\mathbf{x}$ $\Rightarrow$ Coefficients h_k. Input signal $\mathbf{x}$. Instantiated on shift $\mathbf{S}$
- Filter $\mathbf{H}(\hat{\mathbf{S}})\hat{\mathbf{x}}$ $\Rightarrow$ Same Coefficients h_k. Permuted Input signal $\hat{\mathbf{x}}$. Instantiated on permuted shift $\hat{\mathbf{S}}$

Theorem (Permutation equivariance of graph filters)

Consider consistent permutations of the shift operator $\hat{\mathbf{S}} = \mathbf{P}^T\mathbf{S}\mathbf{P}$ and input signal $\hat{\mathbf{x}} = \mathbf{P}^T\mathbf{x}$. Then

$$\mathbf{H}(\hat{\mathbf{S}})\hat{\mathbf{x}} \; = \; \mathbf{P}^T\mathbf{H}(\mathbf{S})\mathbf{x}$$

- Graph filters are equivariant to permutations $\Rightarrow$ Permute input and shift $\equiv$ Permute output

Proof: Write filter output in polynomial form. Use permutation definitions $\hat{\mathbf{S}} = \mathbf{P}^T\mathbf{S}\mathbf{P}$ and $\hat{\mathbf{x}} = \mathbf{P}^T\mathbf{x}$

$$\mathbf{H}(\hat{\mathbf{S}})\hat{\mathbf{x}} = \sum_{k=0}^{K-1} h_k \hat{\mathbf{S}}^k \hat{\mathbf{x}} = \sum_{k=0}^{K-1} h_k \left(\mathbf{P}^T\mathbf{S}\mathbf{P}\right)^k \mathbf{P}^T\mathbf{x}$$

- In the powers $\left(\mathbf{P}^T\mathbf{S}\mathbf{P}\right)^k$, $\mathbf{P}$ and $\mathbf{P}^T$ undo each other $(\mathbf{P}^T\mathbf{P} = \mathbf{I})$ $\Rightarrow \left(\mathbf{P}^T\mathbf{S}\mathbf{P}\right)^k = \mathbf{P}^T\left(\mathbf{S}\right)^k\mathbf{P}$

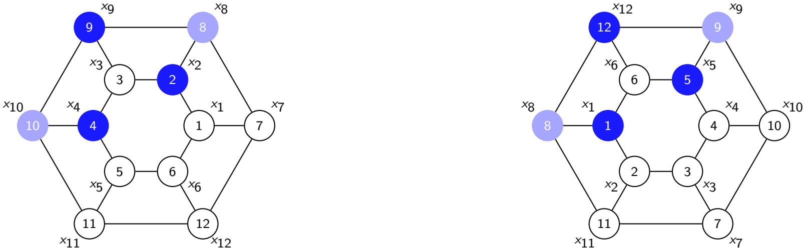
- Substitute this into filter's output expression. Cancel remaining $\mathbf{P}\mathbf{P}^T = \mathbf{I}$ product. Factor $\mathbf{P}^T$

$$\mathbf{H}(\hat{\mathbf{S}})\hat{\mathbf{x}} = \sum_{k=0}^{K-1} h_k \mathbf{P}^T\mathbf{S}^k\mathbf{P}\mathbf{P}^T\mathbf{x} = \sum_{k=0}^{K-1} h_k \mathbf{P}^T\mathbf{S}^k\mathbf{I}\mathbf{x} = \mathbf{P}^T\sum_{k=0}^{K-1} h_k \mathbf{S}^k\mathbf{x} = \mathbf{P}^T\mathbf{H}(\mathbf{S})\mathbf{x} \qquad \blacksquare$$

- We requested signal processing independent of labeling $\Rightarrow$ Graph filters fulfill this request

$\Rightarrow$ Permute input and shift $\equiv$ Relabel input $\Rightarrow$ Permute output $\equiv$ Relabel output

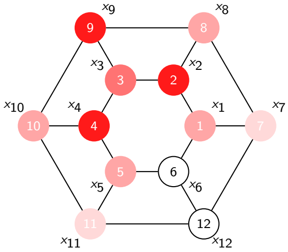
Graph signal $\mathbf{x}$ Supported on $\mathbf{S}$ Graph signal $\hat{\mathbf{x}} = \mathbf{P}^T\mathbf{x}$ supported on $\hat{\mathbf{S}} = \mathbf{P}^T\mathbf{S}\mathbf{P}$



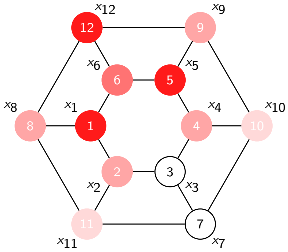
- We requested signal processing independent of labeling $\Rightarrow$ Graph filters fulfill this request

$\Rightarrow$ Permute input and shift $\equiv$ Relabel input $\Rightarrow$ Permute output $\equiv$ Relabel output

Filter's output $\mathbf{H}(\mathbf{S})\mathbf{x}$ Supported on $\mathbf{S}$



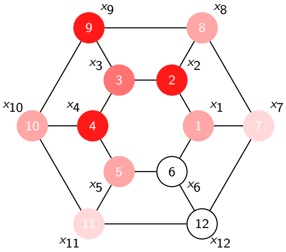
Filter's Output $\mathbf{H}(\hat{\mathbf{S}})\hat{\mathbf{x}}$ supported on $\hat{\mathbf{S}}$



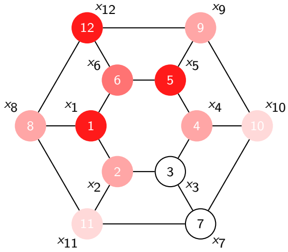
- We requested signal processing independent of labeling $\Rightarrow$ Graph filters fulfill this request

$\Rightarrow$ Permute input and shift $\equiv$ Relabel input $\Rightarrow$ Permute output $\equiv$ Relabel output

Filter's output $\mathbf{H}(\mathbf{S})\mathbf{x}$ Supported on $\mathbf{S}$



Equivariance theorem $\Rightarrow \mathbf{H}(\hat{\mathbf{S}})\hat{\mathbf{x}} = \mathbf{P}^T\mathbf{H}(\mathbf{S})\mathbf{x}$



Permutation Equivariance of Graph Neural Networks

- We will show that graph neural networks inherit the permutation equivariance of graph filters

- L layers recursively process outputs of previous layers. GNN Output parametrized by tensor $\mathcal{H}$

$$\mathbf{x}_\ell = \sigma\left[\sum_{k=0}^{K-1} h_{\ell k}\, \mathbf{S}^k\, \mathbf{x}_{\ell-1}\right] = \sigma\left[\mathbf{H}_\ell(\mathbf{S})\, \mathbf{x}_{\ell-1}\right] \qquad \Phi\Big(\mathbf{x};\ \mathbf{S},\ \mathcal{H}\Big) \ = \ \mathbf{x}_L$$

- We consider running the same GNN on $(\mathbf{S}, \mathbf{x})$ and permuted (relabeled) $(\hat{\mathbf{S}}, \hat{\mathbf{x}}) = (\mathbf{P}^T\mathbf{S}\mathbf{P}, \mathbf{P}^T\mathbf{x})$

$$\Phi\Big(\mathbf{x};\ \mathbf{S},\ \mathcal{H}\Big) \qquad\qquad \Phi\Big(\hat{\mathbf{x}};\ \hat{\mathbf{S}},\ \mathcal{H}\Big)$$

- GNN $\Phi(\mathbf{x}; \mathbf{S}, \mathcal{H}) \ \Rightarrow$ Tensor $\mathcal{H}$. Input signal $\mathbf{x}$. Instantiated on shift $\mathbf{S}$
- GNN $\Phi(\hat{\mathbf{x}}; \hat{\mathbf{S}}, \mathcal{H}) \ \Rightarrow$ Same Tensor $\mathcal{H}$. Permuted Input signal $\hat{\mathbf{x}}$. Instantiated on permuted shift $\hat{\mathbf{S}}$

Theorem (Permutation equivariance of graph neural networks)

Consider consistent permutations of the shift operator $\hat{\mathbf{S}} = \mathbf{P}^T\mathbf{S}\mathbf{P}$ and input signal $\hat{\mathbf{x}} = \mathbf{P}^T\mathbf{x}$. Then

$$\Phi(\hat{\mathbf{x}}; \hat{\mathbf{S}}, \mathcal{H}) \;=\; \mathbf{P}^T\Phi(\mathbf{x}; \mathbf{S}, \mathcal{H})$$

- GNNs equivariant to permutations $\Rightarrow$ Permute input and shift $\equiv$ Permute output

Proof: GNN Layer ℓ recursion on signal $\mathbf{x}_{\ell-1}$ and shift $\mathbf{S}$ $\Rightarrow \mathbf{x}_\ell = \sigma\left[\sum_{k=0}^{K-1} h_{\ell k}\, \mathbf{S}^k\, \mathbf{x}_{\ell-1}\right] = \sigma\left[\mathbf{H}_\ell(\mathbf{S})\mathbf{x}_{\ell-1}\right]$

GNN Layer ℓ recursion on signal $\hat{\mathbf{x}}_{\ell-1}$ and shift $\hat{\mathbf{S}}$ $\Rightarrow \hat{\mathbf{x}}_\ell = \sigma\left[\sum_{k=0}^{K-1} h_{\ell k}\, \hat{\mathbf{S}}^k\, \hat{\mathbf{x}}_{\ell-1}\right] = \sigma\left[\mathbf{H}_\ell(\hat{\mathbf{S}})\hat{\mathbf{x}}_{\ell-1}\right]$

- Assume Layer ℓ inputs satisfy $\hat{\mathbf{x}}_{\ell-1} = \mathbf{P}^T\mathbf{x}_{\ell-1}$. Filters are equivariant. Linearity is pointwise

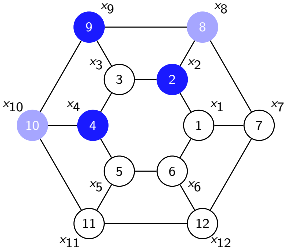
$$\hat{\mathbf{x}}_\ell = \sigma\left[\mathbf{H}_\ell(\hat{\mathbf{S}})\hat{\mathbf{x}}_{\ell-1}\right] = \sigma\left[\mathbf{P}^T\mathbf{H}_\ell(\mathbf{S})\mathbf{x}_{\ell-1}\right] = \mathbf{P}^T\sigma\left[\mathbf{H}_\ell(\mathbf{S})\mathbf{x}_{\ell-1}\right] = \mathbf{P}^T\mathbf{x}_\ell$$

- This in an induction step At Layer 1 we have $\hat{\mathbf{x}} = \mathbf{P}^T\mathbf{x}$ by hypothesis. Induction is complete. ■

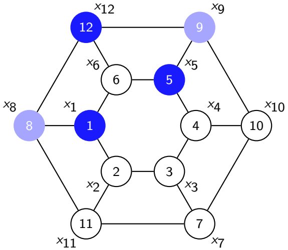
- GNNs, same as graph filters, perform label-independent processing. The nonlinearity is pointwise

$\Rightarrow$ Permute input and shift $\equiv$ Relabel input $\Rightarrow$ Permute output $\equiv$ Relabel output

Graph signal $\mathbf{x}$ Supported on $\mathbf{S}$



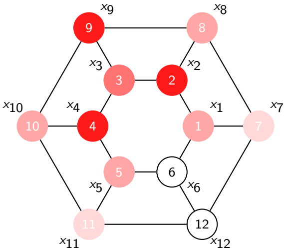
Graph signal $\hat{\mathbf{x}} = \mathbf{P}^T\mathbf{x}$ supported on $\hat{\mathbf{S}} = \mathbf{P}^T\mathbf{S}\mathbf{P}$



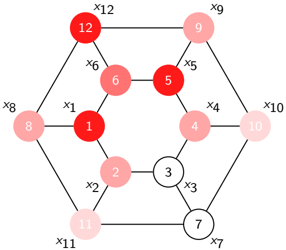
- GNNs, same as graph filters, perform label-independent processing. The nonlinearity is pointwise

$\Rightarrow$ Permute input and shift $\equiv$ Relabel input $\Rightarrow$ Permute output $\equiv$ Relabel output

GNN output $\Phi(\mathbf{x}; \mathbf{S}, \mathcal{H})$ supported on $\mathbf{S}$

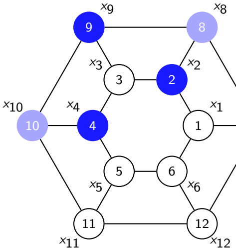


GNN $\Phi(\hat{\mathbf{x}}; \hat{\mathbf{S}}, \mathcal{H}) = \mathbf{P}^T \Phi(\mathbf{x}; \mathbf{S}, \mathcal{H})$ on $\hat{\mathbf{S}} = \mathbf{P}^T \mathbf{S} \mathbf{P}$

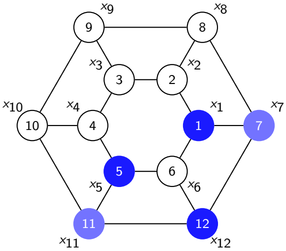


- Equivariance to permutations allows GNNs to exploit symmetries of graphs and graph signals
- By symmetry we mean that the graph can be permuted onto itself $\Rightarrow \mathbf{S} = \mathbf{P}^T\mathbf{S}\mathbf{P}$
- Equivariance theorem implies $\Rightarrow \Phi\Big(\mathbf{P}^T\mathbf{x};\, \mathbf{S}, \mathcal{H}\Big) = \Phi\Big(\mathbf{P}^T\mathbf{x};\, \mathbf{P}^T\mathbf{S}\mathbf{P}, \mathcal{H}\Big) = \mathbf{P}^T\Phi\Big(\mathbf{x};\, \mathbf{S}, \mathcal{H}\Big)$

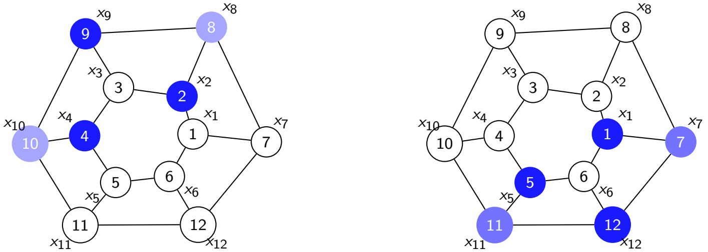
From observing $\mathbf{x}$ supported on $\mathbf{S}$



Learn to process $\mathbf{P}^T\mathbf{x}$ supported on $\mathbf{S} = \mathbf{P}^T\mathbf{S}\mathbf{P}$



- Graph not symmetric but close to symmetric $\Rightarrow$ perturbed version of a permutation of itself



- We will show conditions for stability to deformations $\Rightarrow$ Approximate (close to) equivariance

Definition (Operator Distance Modulo Permutation)

For operators Ψ and $\hat{\Psi}$, the operator distance modulo permutation is defined as

$$\left\|\Psi - \hat{\Psi}\right\|_{\mathcal{P}} = \min_{\mathbf{P}\in\mathcal{P}} \max_{\mathbf{x}:\|\mathbf{x}\|=1} \left\| \mathbf{P}^T\Psi(\mathbf{x}) - \hat{\Psi}(\mathbf{P}^T\mathbf{x}) \right\|$$

where $\mathcal{P}$ is the set of $n \times n$ permutation matrices and where $\|\cdot\|$ stands for the ℓ_2-norm.

- Equivariance to permutations of graph filters $\Rightarrow$ If $\left\|\hat{\mathbf{S}} - \mathbf{S}\right\|_{\mathcal{P}} = \mathbf{0}$. Then $\left\|\mathbf{H}(\hat{\mathbf{S}}) - \mathbf{H}(\mathbf{S})\right\|_{\mathcal{P}} = \mathbf{0}$
- Equivariance to permutations GNNs $\Rightarrow$ If $\left\|\hat{\mathbf{S}} - \mathbf{S}\right\|_{\mathcal{P}} = \mathbf{0}$. Then $\left\|\Phi(\cdot;\hat{\mathbf{S}},\mathcal{H}) - \Phi(\cdot;\mathbf{S},\mathcal{H})\right\|_{\mathcal{P}} = \mathbf{0}$
- When distance $\left\|\hat{\mathbf{S}} - \mathbf{S}\right\|_{\mathcal{P}}$ is small? (not zero) $\Rightarrow$ Stability properties of graph filters and GNNs

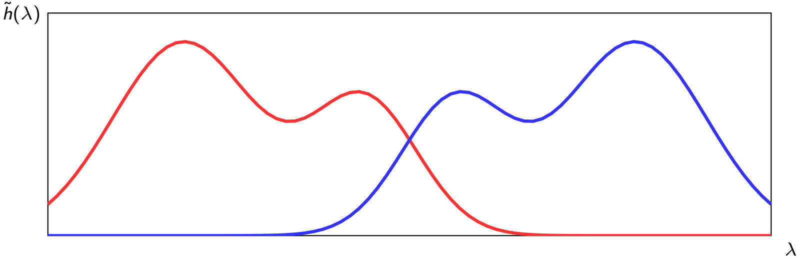
Lipschitz and Integral Lipschitz Filters

- Classes of filters to study discriminablity of GNNs $\Rightarrow$ Lipschitz and integral Lipschitz graph filters

- Graph filters are polynomials on shift operators $\mathbf{S}$ with given coefficients $h_k \Rightarrow \mathbf{H}(\mathbf{S}) = \sum_{k=0}^{\infty} h_k \mathbf{S}^k$

- Filter's frequency response is the same polynomial with scalar variable $\lambda \Rightarrow \tilde{h}(\lambda) = \sum_{k=0}^{\infty} h_k \lambda^k$

- Frequency response determined by filter coefficients h_k. Independent of particular given graph



Definition (Lipschitz Filter)

Given a graph filter with coefficients $\mathbf{h} = \{h_k\}_{k=1}^{\infty}$, and graph frequency response

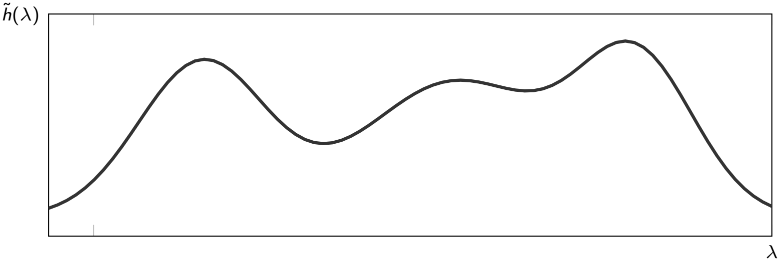
$$\tilde{h}(\lambda) = \sum_{k=0}^{\infty} h_k \lambda^k,$$

we say that the filter is Lipschitz if there exists a constant $C > 0$ such that for λ_1 and λ_2

$$\left| \tilde{h}(\lambda_2) - \tilde{h}(\lambda_1) \right| \leq C \left| \lambda_2 - \lambda_1 \right|.$$

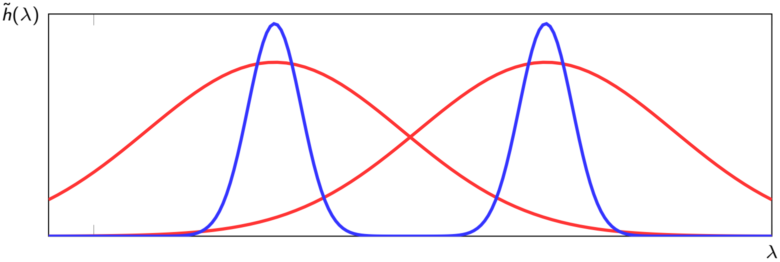
- Change in values of frequency response is at most linear with rate C $\Rightarrow$ Derivative $\tilde{h}'(\lambda) \leq C$

- Frequency response $\tilde{h}(\lambda)$ of Lipschitz filter is Lipschitz continuous $\Rightarrow$ Maximum slope is $\tilde{h}'(\lambda) \leq C$



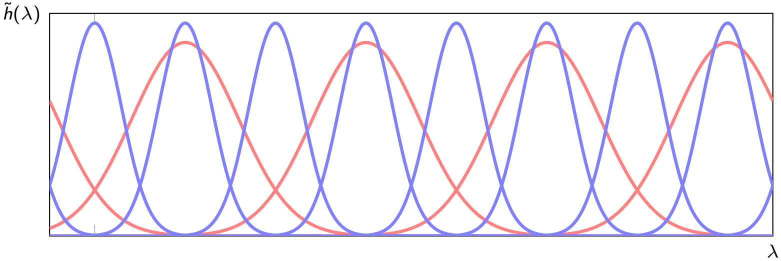
- Lipschitz constant determines discriminability $\Rightarrow$ Small / Large C $\equiv$ Low / High discriminability

- Frequency response $\tilde{h}(\lambda)$ of Lipschitz filter is Lipschitz continuous $\Rightarrow$ Maximum slope is $\tilde{h}'(\lambda) \leq C$



- Lipschitz constant determines discriminability $\Rightarrow$ Small / Large C $\equiv$ Low / High discriminability

- A Lipschitz frame with constant C is made up of Lipschitz filters with constant C
- Larger C allows for sharper filters, that can discriminate more signals. Tighter packing
- The discriminability of the frame is (or can be) the same at all frequencies.



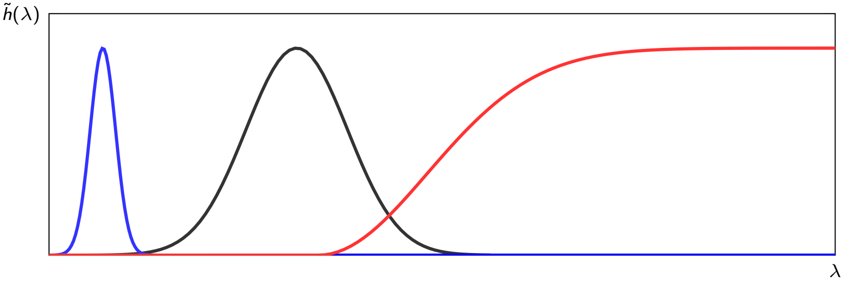
Definition (Integral Lipschitz Filter)

Consider graph filter with coefficients h_k and graph frequency response $\tilde{h}(\lambda) = \sum_{k=0}^{\infty} h_k \lambda^k$. The filter is said integral Lipschitz if there exists constant $C > 0$ such that for all λ_1 and λ_2,

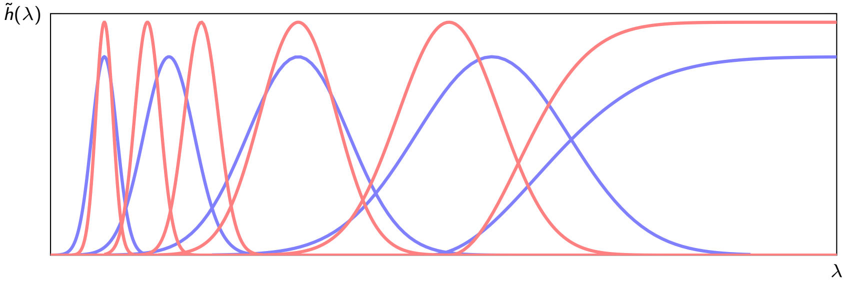
$$|\tilde{h}(\lambda_2) - \tilde{h}(\lambda_1)| \leq C \frac{|\lambda_2 - \lambda_1|}{|\lambda_1 + \lambda_2|/2}.$$

- Lipschitz with a constant that is inversely proportional to the interval's midpoint $\Rightarrow 2C/|\lambda_1 + \lambda_2|$.
- Letting $\lambda_2 \to \lambda_1$ we get that $\lambda \tilde{h}'(\lambda) \leq C$ $\Rightarrow$ The filter can't change for large λ.

- At medium frequencies, integral Lipschitz filters are akin to Lipschitz filters. Roughly speaking
- At low frequencies integral Lipschitz filters can be arbitrarily thin $\Rightarrow$ arbitrary discriminability
- At high frequencies integral Lipschitz filters have to be flat $\Rightarrow$ They lose discriminability



- As Lipschitz frames, integral Lipschitz frames are more discriminative for larger C. Tighter packing
- Except that around $\lambda = 0$, filters can be thin no matter C $\Rightarrow$ High discriminability
- But for large λ filters have to be wide no matter C $\Rightarrow$ No discriminability



Stability of Graph Filters to Scaling

- Scaling of shift operators is a perturbation form that illustrates proof techniques and insights

- We show that graph filters are stable with respect to scaling

Scaling of Shift Operators

- Graphs are subject to estimation error and changes $\Rightarrow$ Running filters on similar graphs
- We scale edges by $(1+\epsilon)$. Scaling deformation of the shift operator $\Rightarrow \hat{\mathbf{S}} = (1+\epsilon)\,\mathbf{S}$
- Deformation model is reasonable $\Rightarrow$ Edges change proportional to their values
- Also unrealistic $\Rightarrow$ All of the edges change by the same proportion

  $\Rightarrow$ Illuminating for discussions. Stability proof contains essential arguments of more generic proof.

Theorem (Integral Lipschitz Graph Filters are Stable to Scaling)

Given graph shift operators $\mathbf{S}$ and $\hat{\mathbf{S}} = (1+\epsilon)\,\mathbf{S}$ and an integral Lipschitz filter with constant C.

The operator norm difference between filters $\mathbf{H}(\mathbf{S})$ and $\mathbf{H}(\hat{\mathbf{S}})$ is bounded as

$$\left\| \mathbf{H}(\hat{\mathbf{S}}) - \mathbf{H}(\mathbf{S}) \right\| \leq C\,\epsilon + \mathcal{O}(\epsilon^2).$$

- Stability to scaling is possible. $\Rightarrow$ But it requires a restriction to the use of integral Lipschitz filters.

- The key arguments of the proof are in the GFT domain. We provide two preliminary spectral facts.

Fact 1:

If $\tilde{\mathbf{x}} = \mathbf{V}^H\mathbf{x}$ is the GFT of $\mathbf{x}$ we can write $\Rightarrow \mathbf{x} = \sum_{i=1}^{n} \tilde{x}_i \mathbf{v}_i,$ where $\mathbf{v}_i$ are the eigenvectors of $\mathbf{S}$

Proof: Write $\mathbf{x}$ using the inverse GFT $\Rightarrow \mathbf{x} = \mathbf{V}\tilde{\mathbf{x}} = \left[\mathbf{v}_1, \ldots, \mathbf{v}_n\right] \times \begin{bmatrix} \tilde{x}_1 \\ \vdots \\ \tilde{x}_n \end{bmatrix} = \tilde{x}_1\mathbf{v}_1 + \ldots + \tilde{x}_n\mathbf{v}_n$

- The key arguments of the proof are in the GFT domain. We provide two preliminary spectral facts.

Fact 2:

The frequency response derivative is $\tilde{h}'(\lambda) = \sum_{k=0}^{\infty} k\, h_k\, \lambda^{k-1}$. Consequently $\lambda \tilde{h}'(\lambda) = \sum_{k=0}^{\infty} k\, h_k\, \lambda^{k}$.

Proof: Frequency response is the series $\Rightarrow \tilde{h}(\lambda) = \sum_{k=0}^{\infty} h_k\, \lambda^k$. The summands' derivatives are $k\, h_k\, \lambda^{k-1}$.

Proof: Filter difference given by graph filter definition $\mathbf{H}(\mathbf{S}) = \sum_{k=0}^{\infty} h_k \mathbf{S}^k$. Further write $\hat{\mathbf{S}} = (1+\epsilon)\,\mathbf{S}$

$$\mathbf{H}(\hat{\mathbf{S}}) - \mathbf{H}(\mathbf{S}) = \sum_{k=0}^{\infty} h_k \hat{\mathbf{S}}^k - \sum_{k=0}^{\infty} h_k \mathbf{S}^k = \sum_{k=0}^{\infty} h_k \Big[\big((1+\epsilon)\,\mathbf{S} \big)^k - \hat{\mathbf{S}}^k \Big]$$

- Expand binomial $\big((1+\epsilon)\,\mathbf{S}\big)^k$ to first order only. Group all high order terms in matrix $\mathbf{O}_k(\epsilon)$

$$\big((1+\epsilon)\,\mathbf{S} \big)^k = (1+k\epsilon)\,\mathbf{S}^k + \mathbf{O}_k(\epsilon)$$

- Upon substitution the terms $\mathbf{S}^k$ cancel out $\Rightarrow \mathbf{H}(\hat{\mathbf{S}}) - \mathbf{H}(\mathbf{S}) = \sum_{k=0}^{\infty} h_k k \epsilon \mathbf{S}^k + \mathbf{O}(\epsilon)$

- The matrix $\mathbf{O}(\epsilon)$ satisfies $0 < \lim_{\epsilon \to 0} \frac{\|\mathbf{O}(\epsilon)\|}{\epsilon^2} < \infty$ because filter is analytic. Term is of order $\mathcal{O}(\epsilon^2)$

Proof Step 2: Evaluating and Reducing the Operator Norm

- Have reduced the filter difference to $\Rightarrow \mathbf{H}(\hat{\mathbf{S}}) - \mathbf{H}(\mathbf{S}) \; = \; \sum_{k=0}^{\infty} h_k k \epsilon \mathbf{S}^k \; + \; \mathbf{O}(\epsilon) \; = \; \boldsymbol{\Delta}(\mathbf{S}) + \mathbf{O}(\epsilon)$

- Where we have defined the filter variation $\boldsymbol{\Delta}(\mathbf{S}) = \epsilon \sum_{k=0}^{\infty} k h_k \mathbf{S}^k$ to simplify notation

- Triangle inequality $\Rightarrow \|\mathbf{H}(\hat{\mathbf{S}}) - \mathbf{H}(\mathbf{S})\| \; \leq \; \|\boldsymbol{\Delta}(\mathbf{S})\| + \mathbf{O}(\epsilon) \; = \; \|\boldsymbol{\Delta}(\mathbf{S})\| + \mathcal{O}(\epsilon^2)$

- Since $\|\boldsymbol{\Delta}(\mathbf{S})\| = \max_{\|\mathbf{x}\|=1} \|\boldsymbol{\Delta}(\mathbf{S})\mathbf{x}\|$ theorem follows if we prove $\|\boldsymbol{\Delta}(\mathbf{S})\mathbf{x}\| \leq C\epsilon$ for all $\mathbf{x}$ with $\|\mathbf{x}\| = 1$

Proof Step 3: Shifting to the GFT Domain

- Product of filter variation with unit norm $\mathbf{x}$. Write the iGFT of the input $\mathbf{x} = \sum_{i=1}^{n} \tilde{x}_i \mathbf{v}_i \quad (\mathbf{S}\mathbf{v}_i = \lambda_i \mathbf{v}_i)$

$$\mathbf{\Delta}(\mathbf{S})\,\mathbf{x} \;=\; \epsilon \sum_{k=0}^{\infty} k\, h_k\, \mathbf{S}^k\, \mathbf{x} \;=\; \epsilon \sum_{k=0}^{\infty} k\, h_k\, \mathbf{S}^k \times \left[\sum_{i=1}^{n} \tilde{x}_i \mathbf{v}_i\right] \;=\; \sum_{i=1}^{n} \tilde{x}_i\, \epsilon \sum_{k=0}^{\infty} k\, h_k\, \mathbf{S}^k\, \mathbf{v}_i$$

- Since the $\mathbf{v}_i$ are eigenvectors of $\mathbf{S}$ $\Rightarrow \mathbf{S}^k \mathbf{v}_i = \lambda_i^k \mathbf{v}_i$. With λ_i the associated eigenvalue

$$\mathbf{\Delta}(\mathbf{S})\mathbf{x} \;=\; \epsilon \sum_{i=1}^{n} \tilde{x}_i \sum_{k=0}^{\infty} k\, h_k\, \mathbf{S}^k \mathbf{v}_i \;=\; \epsilon \sum_{i=1}^{n} \tilde{x}_i \sum_{k=0}^{\infty} k\, h_k\, \lambda_i^k \mathbf{v}_i \;=\; \epsilon \sum_{i=1}^{n} \tilde{x}_i \sum_{k=0}^{\infty} k\, h_k\, \lambda_i^k \mathbf{v}_i \;=\; \epsilon \sum_{i=1}^{n} \tilde{x}_i \sum_{k=0}^{\infty} k\, h_k\, \lambda_i^k \mathbf{v}_i \;=$$

- The derivative of the filter's response appears $\Rightarrow \sum_{k=0}^{\infty} k\, h_k\, \lambda_i^k \;=\; \lambda_i\, \tilde{h}'(\lambda_i)$

- End up with remarkably simple equation $\Rightarrow \mathbf{\Delta}(\mathbf{S})\mathbf{x} = \epsilon \sum_{i=1}^{n} \tilde{x}_i \sum_{k=0}^{\infty} k\, h_k\, \lambda_i^k \mathbf{v}_i = \epsilon \sum_{i=1}^{n} \tilde{x}_i \Big(\lambda_i\, \tilde{h}'(\lambda_i) \Big) \mathbf{v}_i$

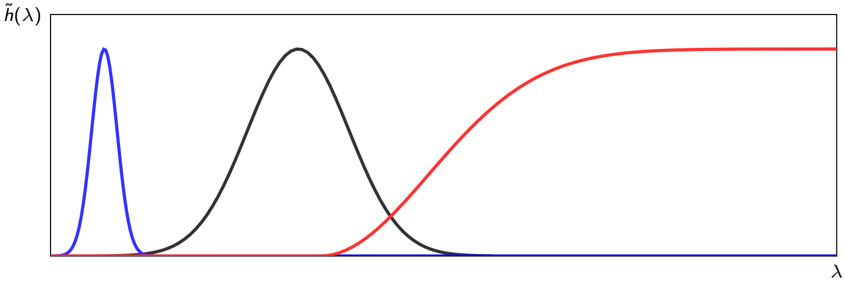
- Which involves the quantity we bound with the integral Lipschitz condition $\Rightarrow \left|\lambda_i\, \tilde{h}'(\lambda_i)\right| \leq C$

- Compute energy. Use integral Lipschitz bound. Recall that signal has unit energy, $\|\mathbf{x}\|^2 = \|\tilde{\mathbf{x}}\|^2 = 1$

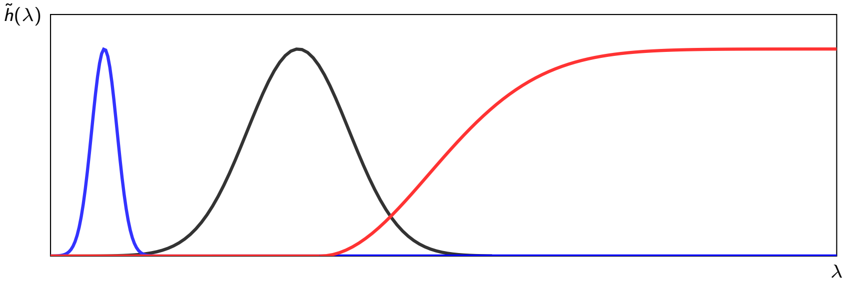
$$\left\|\mathbf{\Delta}(\mathbf{S})\mathbf{x}\right\|^2 \;=\; \epsilon^2 \sum_{i=1}^{n} \tilde{x}_i^2 \Big(\lambda_i\, \tilde{h}'(\lambda_i) \Big)^2 \leq \epsilon^2 \sum_{i=1}^{n} \tilde{x}_i^2\, C^2 \;=\; (C\epsilon)^2$$

- Take square root ■

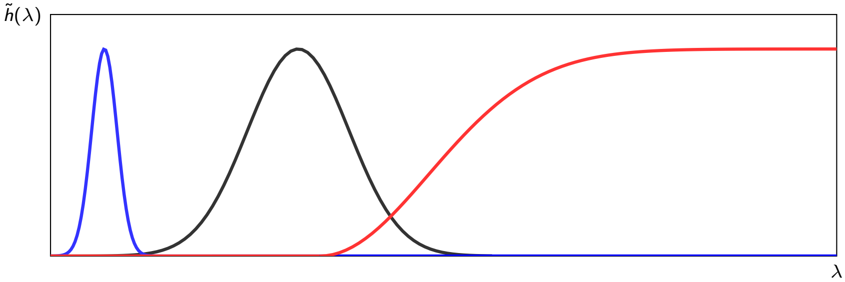
- Integral Lipschitz filters are necessary for stability to deformations of the supporting graph
- This is not an artifact of the analysis. The result is tight. The term $\sum_{k=0}^{\infty} k\, h_k\, \lambda_i^k = \lambda_i\, h'(\lambda_i)$ appears.



- One would expect a stability vs discriminability tradeoff. But in a sense, we get a non-tradeoff.
- Integral Lipschitz filters have to be flat at high frequencies. $\Rightarrow$ They can't discriminate
- It is impossible to separate signals with high frequency features and be stable to deformations



- One would expect a stability vs discriminability tradeoff. But in a sense, we get a non-tradeoff.
- Integral Lipschitz filters have to be flat at high frequencies. $\Rightarrow$ They can't discriminate
- It is impossible to separate signals with high frequency features and be stable to deformations



Stability of Graph Neural Networks to Scaling

- Scaling of shift operators is a perturbation form that illustrates proof techniques and insights
- We show that Graph Neural Networks are stable with respect to scaling

- To avoid appearance of meaningless constants we normalize the filters and the nonlinearity.

- At each layer of the GNN, the filters have unit operator norm $\Rightarrow \|\mathbf{H}_\ell(\mathbf{S})\| = 1$

  $\Rightarrow$ Easy to achieve with scaling $\Rightarrow$ Equivalent to $\max_\lambda \tilde{h}_\ell(\lambda) = 1$

- The nonlinearity σ is Lipschitz and normalized so that $\Rightarrow \|\sigma(\mathbf{x}_2) - \sigma(\mathbf{x}_1)\| \leq \|\mathbf{x}_2 - \mathbf{x}_1\|$

  $\Rightarrow$ Easy to achieve with scaling. True of ReLU, hyperbolic tangent, and absolute value

- Joining both assumptions $\Rightarrow$ If input energy is $\|\mathbf{x}\| \leq 1$, all layer outputs have energy $\|\mathbf{x}_\ell\| \leq 1$

Theorem (Integral Lipschitz GNNs are Stable to Scaling)

Given shift operators $\mathbf{S}$ and $\hat{\mathbf{S}} = (1+\epsilon)\,\mathbf{S}$ and a GNN operator $\Phi(\cdot;\mathbf{S},\mathcal{H})$ with L single-feature layers. The filters at each layer have unit operator norms and are integral Lipschitz with constant C. The nonlinearity σ is normalized Lipschitz. Then

$$\left\| \Phi(\cdot;\mathbf{S},\mathcal{H}) - \Phi(\cdot;\hat{\mathbf{S}},\mathcal{H}) \right\| \leq C\,L\,\epsilon + \mathcal{O}(\epsilon^2).$$

- GNNs inherit the stability of graph filters. It's the same bound. Propagated through L layers

Proof: The theorem is true because the nonlinearity is pointwise. It is unaware of the graph.

- Formally $\Rightarrow$ Let $\mathbf{x}_\ell$ be the Layer ℓ output of GNN $\Phi(\mathbf{x}; \mathbf{S}, \mathcal{H})$

  $\Rightarrow$ Let $\hat{\mathbf{x}}_\ell$ be the Layer ℓ output of GNN $\Phi(\hat{\mathbf{x}}; \hat{\mathbf{S}}, \mathcal{H})$

- Layer ℓ is a perceptron with filter $\mathbf{H}_\ell$ $\Rightarrow$ $\left\| \mathbf{x}_\ell - \hat{\mathbf{x}}_\ell \right\| = \left\| \sigma\Big[\mathbf{H}_\ell(\mathbf{S})\mathbf{x}_{\ell-1}\Big] - \sigma\Big[\mathbf{H}_\ell(\hat{\mathbf{S}})\hat{\mathbf{x}}_{\ell-1}\Big] \right\|$

- Nonlinearity is normalized Lipschitz $\Rightarrow$ $\left\| \mathbf{x}_\ell - \hat{\mathbf{x}}_\ell \right\| \leq \left\| \mathbf{H}_\ell(\mathbf{S})\mathbf{x}_{\ell-1} - \mathbf{H}_\ell(\hat{\mathbf{S}})\hat{\mathbf{x}}_{\ell-1} \right\|$

- This is the critical step of the proof. The rest of the proof is just algebra.

Proof Step 2: Implementing Norm Manipulations

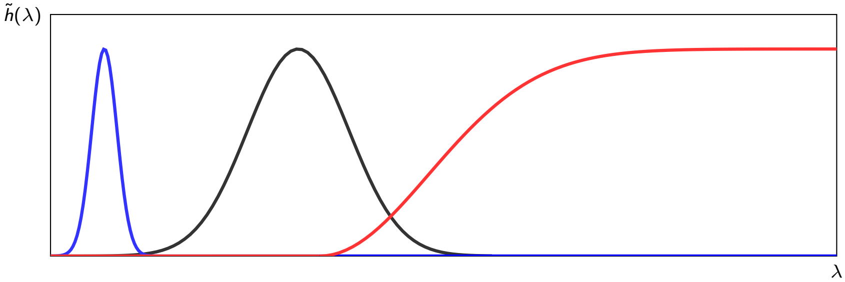
- In last bound, add and subtract $\mathbf{H}_\ell(\hat{\mathbf{S}})\mathbf{x}_{\ell-1}$. Triangle inequality. Submultiplicative property of norms

$$\left\| \mathbf{x}_\ell - \hat{\mathbf{x}}_\ell \right\| \leq \left\| \mathbf{H}_\ell(\mathbf{S})\mathbf{x}_{\ell-1} - \mathbf{H}_\ell(\hat{\mathbf{S}})\hat{\mathbf{x}}_{\ell-1} + \mathbf{H}_\ell(\hat{\mathbf{S}})\mathbf{x}_{\ell-1} - \mathbf{H}_\ell(\hat{\mathbf{S}})\mathbf{x}_{\ell-1} \right\|$$

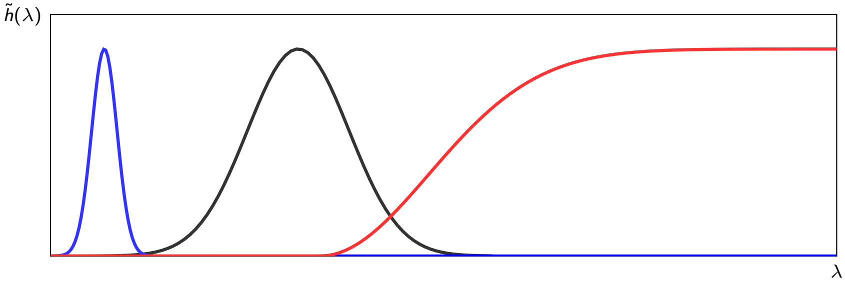
$$\leq \left\| \mathbf{H}_\ell(\mathbf{S}) - \mathbf{H}_\ell(\hat{\mathbf{S}}) \right\| \times \left\| \mathbf{x}_{\ell-1} \right\| + \left\| \mathbf{H}_\ell(\hat{\mathbf{S}}) \right\| \times \left\| \mathbf{x}_{\ell-1} - \hat{\mathbf{x}}_{\ell-1} \right\|$$

- Since filters are normalized $\Rightarrow$ Filter norm $\left\| \mathbf{H}_\ell(\hat{\mathbf{S}}) \right\| = 1$. Signal norm $\Rightarrow \left\| \mathbf{x}_{\ell-1} \right\| \leq 1$
- The theorem on stability of filters to scaling holds $\Rightarrow \left\| \mathbf{H}_\ell(\mathbf{S}) - \mathbf{H}_\ell(\hat{\mathbf{S}}) \right\| \leq \epsilon C + \mathcal{O}(\epsilon^2)$
- Put all bounds together $\Rightarrow \left\| \mathbf{x}_\ell - \hat{\mathbf{x}}_\ell \right\| \leq \epsilon C \times 1 + 1 \times \left\| \mathbf{x}_{\ell-1} - \hat{\mathbf{x}}_{\ell-1} \right\| + \mathcal{O}(\epsilon^2)$
- Apply recursively from Layer L back to Layer 1. The L factor appears ■

- GNNs have the same stability properties of graph filters. They need integral Lipschitz filters.
- Integral Lipschitz filters have to be flat at high frequencies. $\Rightarrow$ They can't discriminate
- It is impossible to separate signals with high frequency features and be stable to deformations



- GNNs have the same stability properties of graph filters. They need integral Lipschitz filters.
- On the flip side, integral Lipschitz filter can be very sharp at low frequencies
- We can be very discriminative at low frequencies. And at the same very stable to deformations



- GNNs use low-pass nonlinearities to demodulate high frequencies into low frequencies
- Where they can be discriminated sharply with a stable filter at the next layer
- Thus, they can be stable and discriminative. Something that linear graph filters can't be