Geometric Scattering And Applications

Michael Perlmutter

Department of Mathematics
University of California, Los Angeles

Outline:

1. The Euclidean Scattering Transform
   - A wavelet based model of CNNs
2. Geometric Scattering
   - Wavelets on Graphs and Manifolds
3. Scattering for Graph Synthesis
   - Generate New Molecules

Supervised Learning

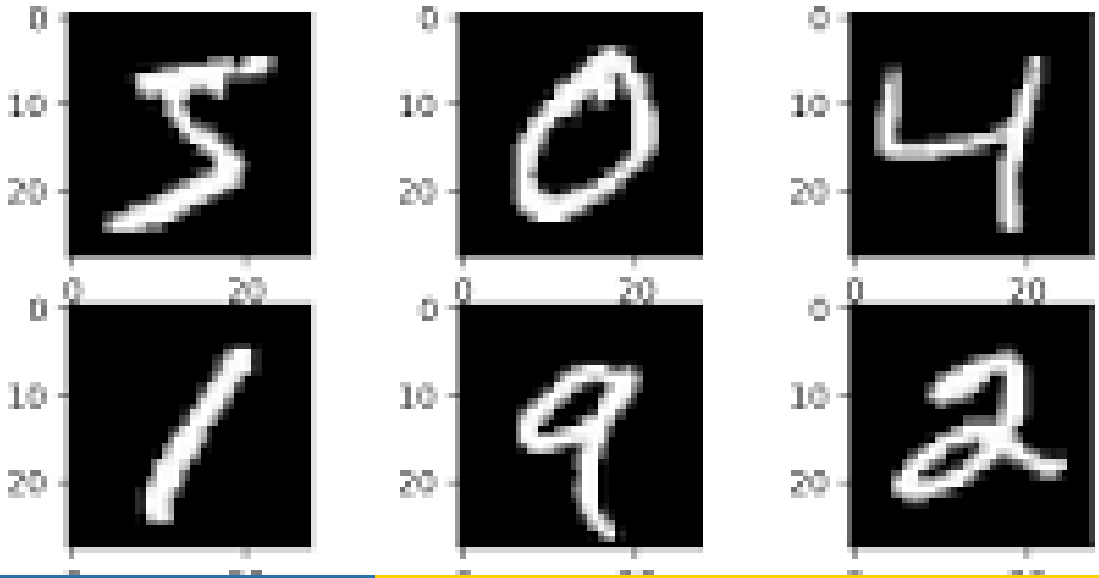
Problem Formulation

Learn an unknown function $f : \mathcal{X} \to \mathcal{Y}$ from pointwise evaluations

$$(x_1, y_1), \ldots, (x_N, y_N)$$

Image Recognition - MNIST data set

$\mathcal{X}$ is the set of all images $\mathcal{Y}$ is the set $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$



Convolutional Neural Networks

Basic Structure

- Front end: Learns a representation of input through many convolutional layers
  - Each convolutional layer consists of a linear transformation and a pointwise nonlinearity, e.g. $\text{ReLU}(x) = \max\{x, 0\}$.
- Back end: Uses this representation to classify the input
- The convolutional layers and the linear classifier are jointly optimized using back propagation.

.

Drawbacks and Challenges

- Interpretability?
- Data Hungry
- Why are many layers better than one gigantic layer?

The (Euclidean) Scattering Transform - S. Mallat (2012)

Overview:

- Model of Convolutional Neural Networks.
- Predefined (wavelet) filters.

Advantages:

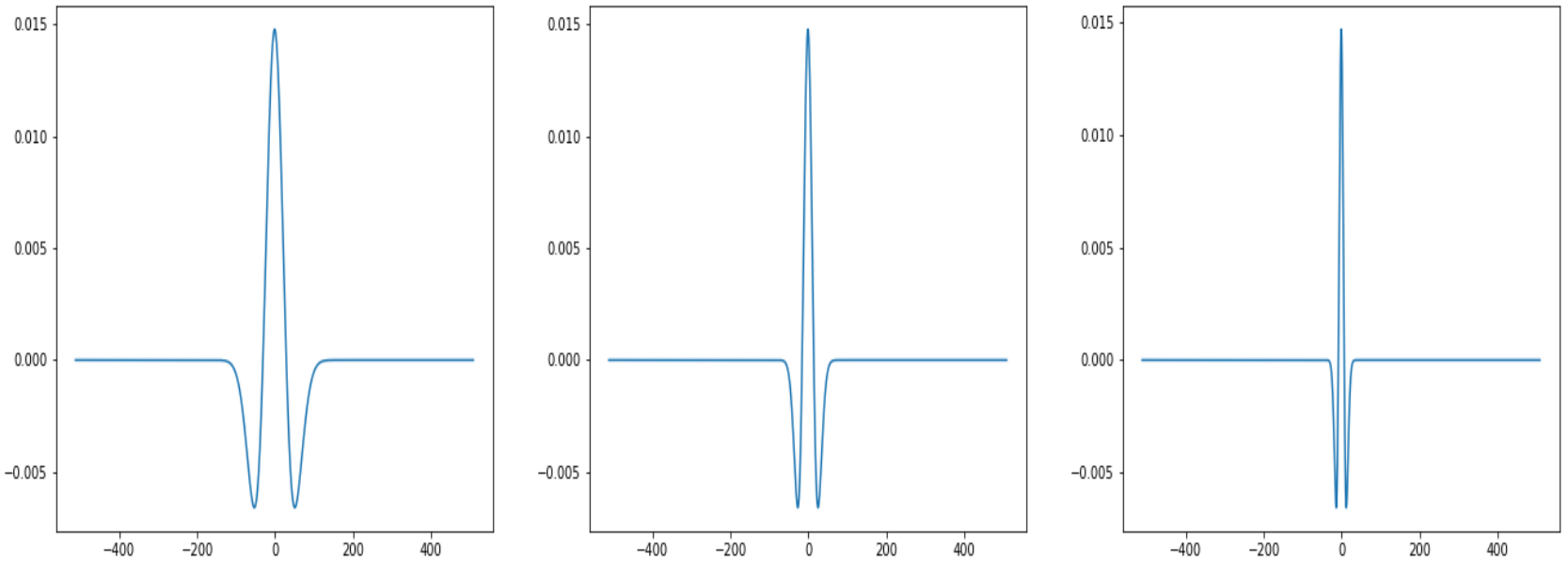
- Provable stability and invariance properties.
- Near state of the art numerical results in certain situations.
- Needs less training data.

Characterizing a function

Collect measurements:

Given a signal $f(x)$, collect measurements that encode information.

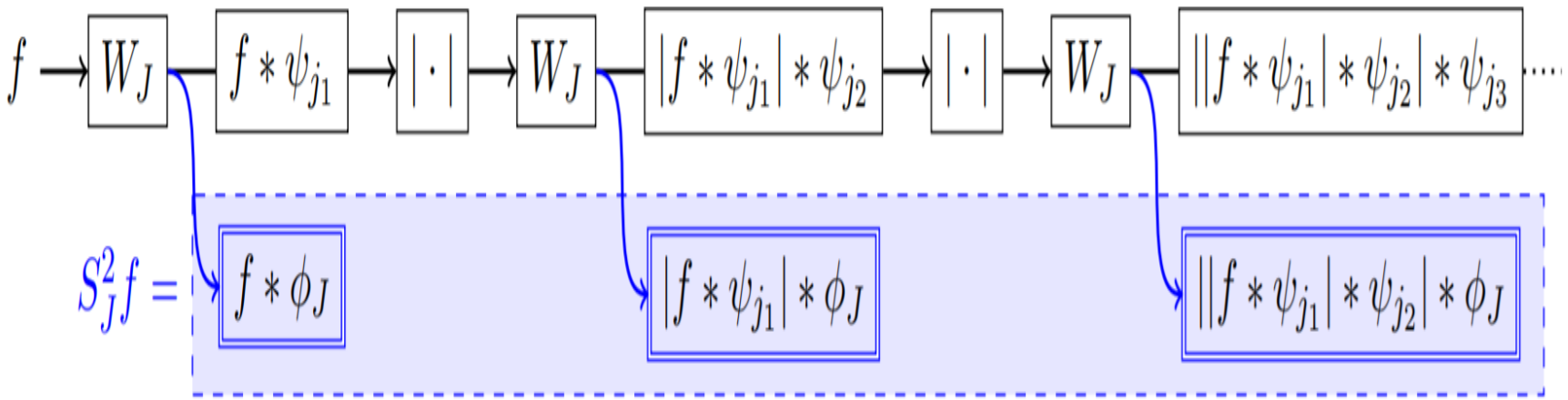
- The Fourier Series: $c_n(f) = \int_0^1 f(x)\mathbb{e}^{-2\pi inx}dx$,
- The Wavelet Transform: $W_j f(x) = (\psi_j \star f)(x)$,
- $\psi_j(x) = \frac{1}{2^j}\psi\left(\frac{x}{2^j}\right)$ for some mean zero "mother wavelet" ψ.



The Scattering Transform

The Scattering Transform:

- Multilayered cascade of nonlinear measurements.
- Each "layer" uses a wavelet transform W_J and a nonlinearity,
- $U_j f(x) = \sigma((\psi_j \star f)(x)),\ j \leq J, \quad \sigma(x) = M(x) = |x|.$
- $U_{j_1,j_2} f(x) = U_{j_2} U_{j_1} f(x)$
- $U_{j_1,\ldots,j_m} f(x) = U_{j_m} \ldots U_{j_1} f(x)$
- $S_{j_1,\ldots j_m} f(x) = \phi_J \star U_{j_1,\ldots,j_m} f(x), \quad \phi_J(x) = \frac{1}{2^J}\phi\left(\frac{x}{2^J}\right)$



Scattering vs. CNNs for Classification

How they work:

- Iterative cascade of convolutions and nonlinearities.
- Scattering uses predesigned wavelet filters. CNNs find their filters by solving a (highly nonconvex) optimization problem.
- Scattering uses $M(x) = |x|$ rather than more common choices such as ReLu.

Situations where scattering is appropriate:

- Limited amounts of (labeled, trustworthy) training data.
- Want to account for the underlying physics.

Why a Nonlinear Structure?

A good representation should be:

- Stable on $\mathbf{L^2}$
- Invariant to translations (or rotations etc.)
- Sufficiently descriptive

The limits of linearity:

A linear network can be invariant or descriptive, but not both.

- $\widehat{f}(0) = \int_{\mathbb{R}^d} f(x)dx$ is invariant, but throws away all high-frequency information.
- Filters which focus in on high-frequency information are unstable to translations.

The wavelet transform captures high-frequency information, and the modulus operator pushes this information down to lower frequencies.

Theoretical Properties

Theorem (Mallat 2012)

The scattering transform has the following properties

- *Nonexpansiveness: i.e..,*

$$\|Sf_1 - Sf_2\| \leq \|f_1 - f_2\|, \quad \forall f_1, f_2 \in \mathbf{L}^2$$

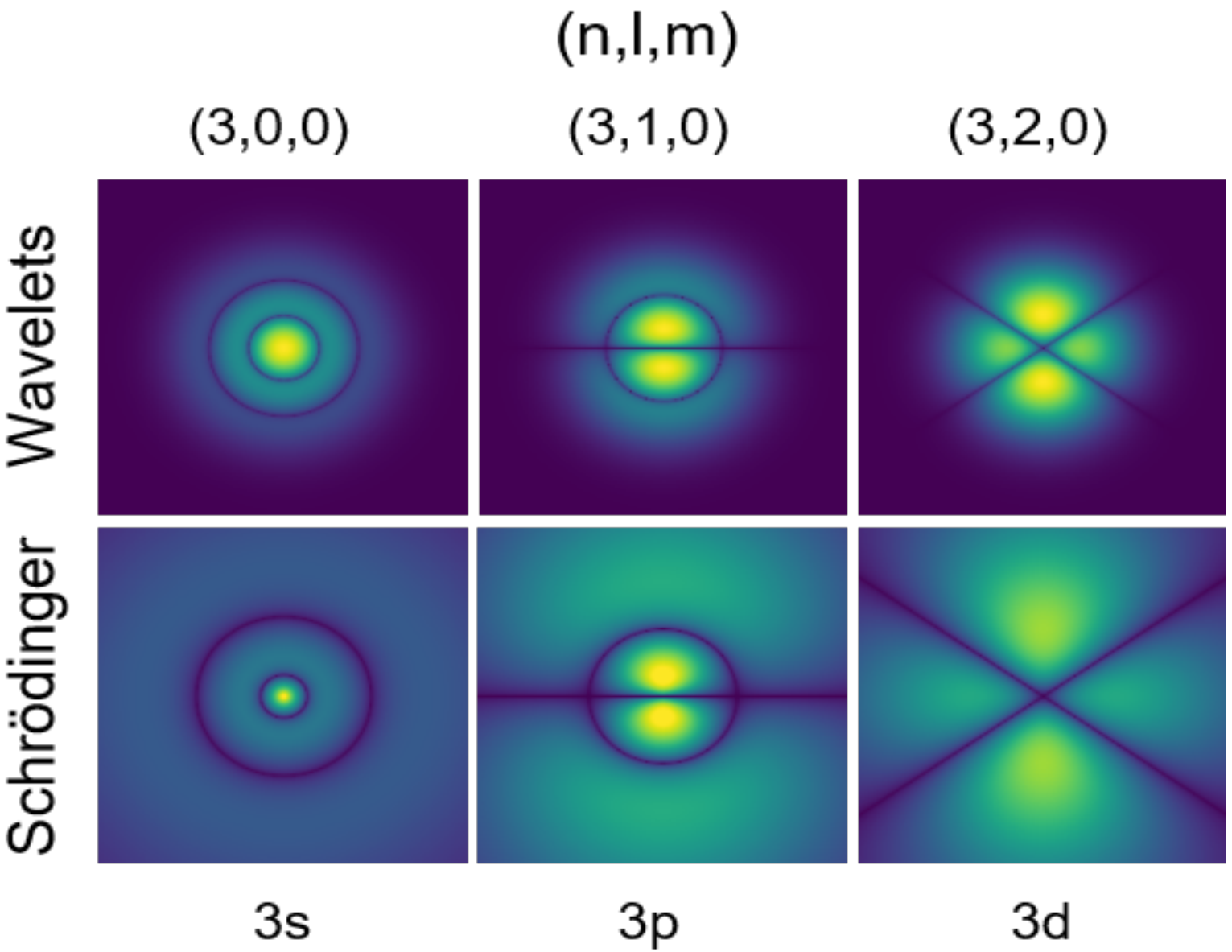
- *Invariance to translations*
- *Stability to small deformations*

Limited Data Environment - Scattering for Stylometry

Which one is a Van Gogh?

- *Scattering Transform and Sparse Linear Classifiers for Art Authentication* (Leonarduzzi, Liu, and Wang)
- Dataset of 64 real Van Gogh's and 15 fakes.
- Scattering achieves state-of-the-art (96%) accuracy.

Scattering for Quantum Chemistry



Same Power Spectrum, Different Scattering

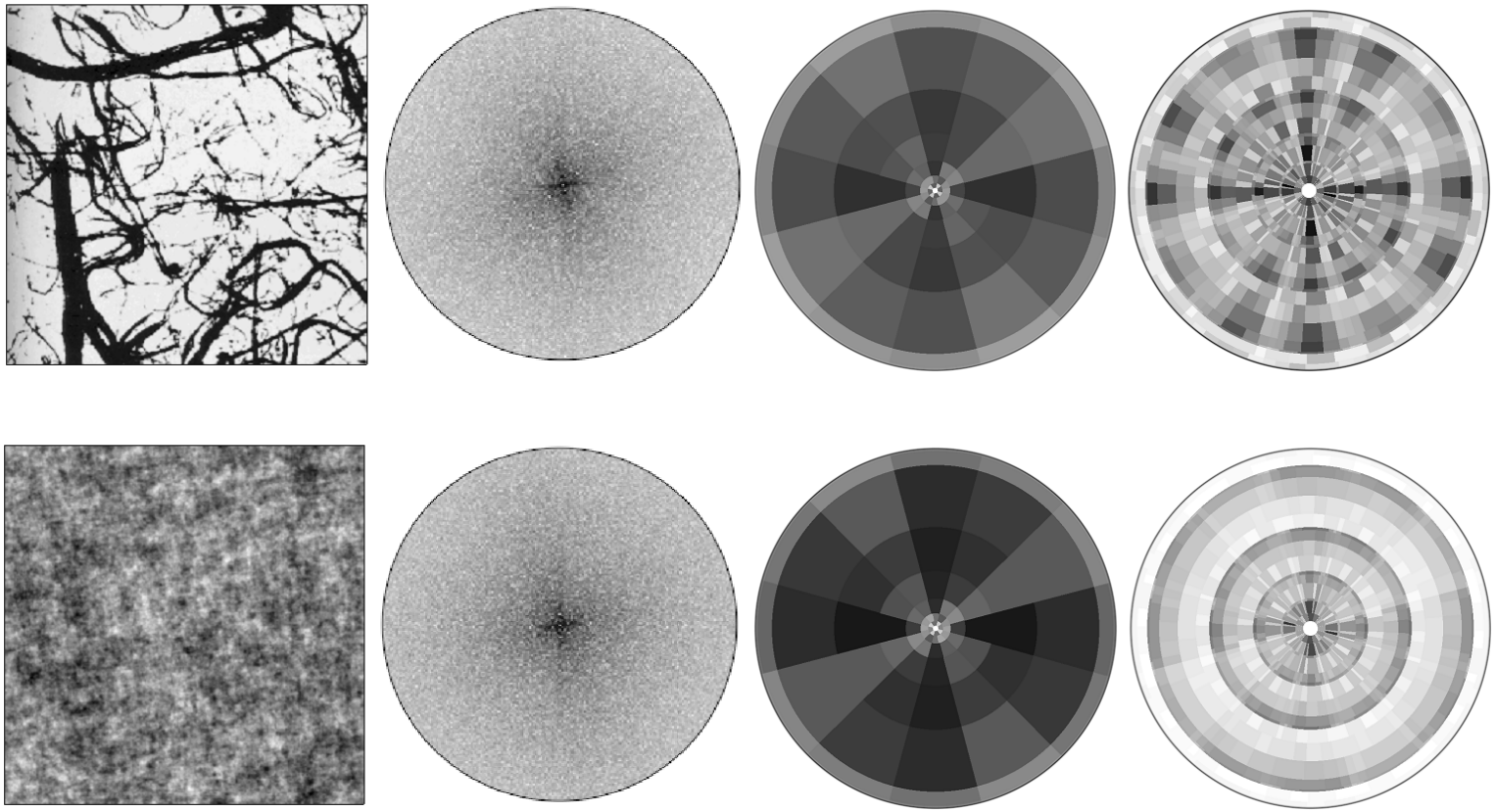


Figure 9: Two different textures having the same Fourier power spectrum. (a) Textures $X(u)$. Top: Brodatz texture. Bottom: Gaussian process. (b) Same estimated power spectrum $\hat{P}X(\omega)$. (c) Nearly same scattering coefficients $S_J[p]X$ for $m = 1$ and 2^J equal to the image width. (d) Different scattering coefficients $S_J[p]X$ for $m = 2$.

Coefficients of Common Stochastic Processes

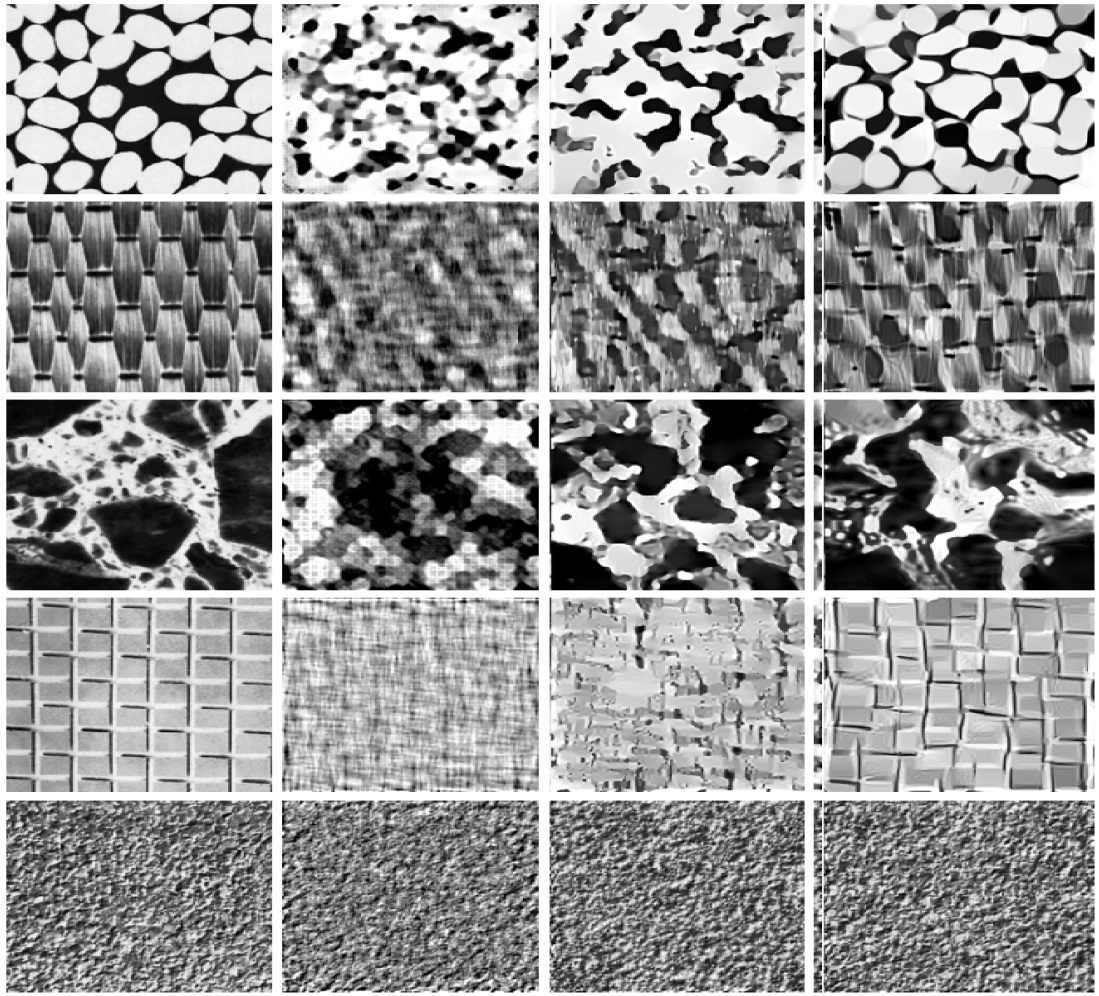
Informal Theorem: (Bruna, Mallat, Bacry, Muzy)

- Bruna et al. compute (in asymptotic limits) the scattering coefficients of common stochastic processes
  - Poisson Process
  - Fractional Brownian motion
  - α-stable
- First-order scattering coefficients can distinguish Poisson vs fractional Brownian motion or α-stable
- Second-order coefficients can distinguish fBM vs stable

Central Limit Theorems

- Works by G.R. Liu, Y.C Sheu, and H.T. Wu prove central limit theorem type results for higher-order moments
- Use more general activation functions

Synthesis of random textures



(a) (b) (c) (d)

(a): Original texture. (b): texture synthesized with wavelet $\mathbf{l^2}$ norms. (c): synthesized with wavelet $\mathbf{l^1}$ norms. (d): synthesized with scattering coefficients.

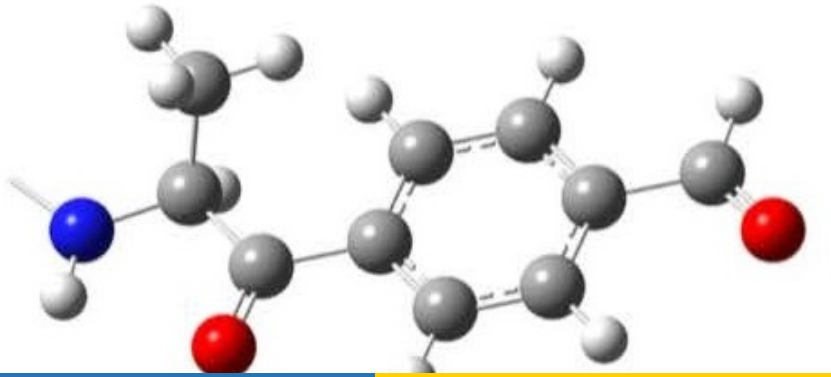
Geometric Deep Learning

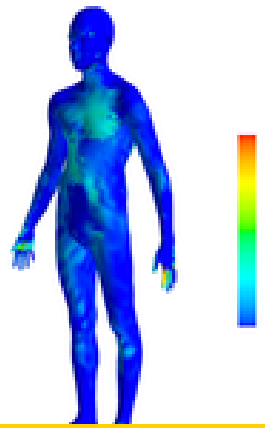
Goal:

- Extend Deep Learning methods to data with non-Euclidean Structure such as graphs and manifolds

Geometric Scattering:

- Key challenge is defining wavelets
- Probabilistic Methods: Heat semi-group on a manifold or random walk on a graph.
- Spectral Methods: Eigenfunctions / eigenvectors of an appropriate Laplacian.





Notation

- $G = (V, E)$ is a graph, $V = \{v_1, \ldots, v_N\}$, $E \subseteq V \times V$
- Adjacency matrix A

$$A(j,k) = \begin{cases} 1 & \text{if } (v_j, v_k) \in E \\ 0 & \text{otherwise} \end{cases}$$

- Degree vector and matrix

$$D = \text{diag}(\mathbf{d}), \quad \mathbf{d}(j) = \text{degree of vertex } j$$

- Lazy Random Walk Matrix $P = I + AD^{-1}$
- Lazy Symmetric Diffusion Matrix $T = I + D^{-1/2}AD^{-1/2}$
- Normalized Self-Loop Adjacency Matrix (GCN)

$$\widehat{A} = (D + I)^{-1/2}(A + I)(D + I)^{-1/2}$$

Semi-Supervised Node Classification

Setup

- Entire Graph Structure is known (all Vertices and Edges)
- Node feature matrix $X = X^0 = (\mathbf{x}_1, \ldots, \mathbf{x}_C)$ is known for all nodes
- Labels are known for some nodes ($\leq$5%)
- Goal: Predict the labels of the remaining nodes.

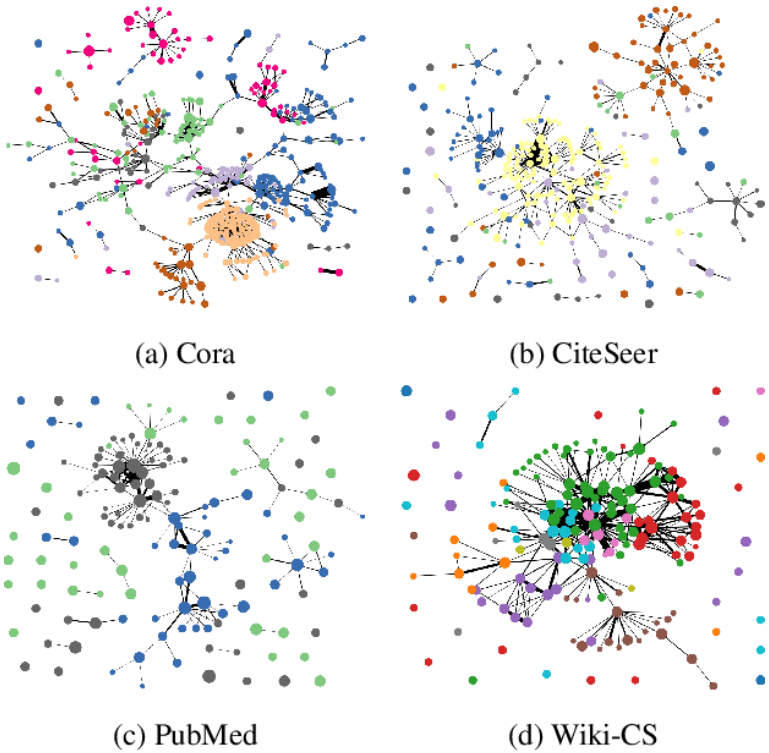


Figure: Visualizations of Common Data sets

Graph Convolutional Network (Kipf and Welling)

Layer-Wise Update Rule

- Sequentially transform node features via layerwise updates

$$X^{t+1} = \sigma(\widehat{A}X^t\Theta)$$

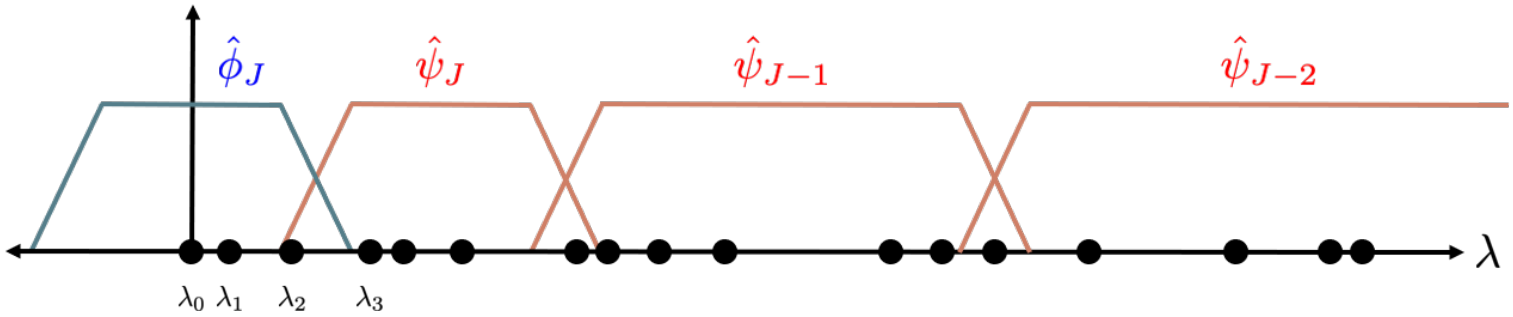
- Θ is a trainable weight matrix.
- The matrix $\widehat{A}$ acts a local-averaging operator.
- Promotes smoothness, i.e. similarity amongst neighbors
- Θ is learned but $\widehat{A}$ is designed.

Low-pass filter

- Multiplying by $\widehat{A}$ leaves bottom eigenvector unchanged.
- All other frequencies are depressed.
- Repeated applications increasingly depress high-frequencies.
- "Deep" Graph Neural Nets typically use 2 layers.

Geometric Wavelets

- Detects changes rather than local-averages
  - How is my four-step neighborhood different than my two-step neighborhood?
- Band-pass filter rather than low-pass



Spatial Geometric Wavelets

Definition

Let $\mathcal{X}$ be a graph or a manifold and let $\{P_t\}_{t\geq 0}$ be the heat-semigroup of random walk diffusion.

$$\mathcal{W}_J^{(2)} f(x) = \{\Psi_j^{(2)} f(x), \Phi_J^{(2)} f(x)\}_{0\leq j\leq J},$$

where

$$\Psi_j^{(2)} = P_{2^{J+1}} - P_{2^J}, \quad \Phi_J^{(2)} = P_{2^{J+1}},$$

Theorem: P., Gao, Wolf, Hirn

$\mathcal{W}_J^{(2)}$ is a non-expansive frame on a suitable weighted space, i.e.,

$$c\|f\|^2 \leq \sum_j \|\Psi_j^{(2)} f\|^2 + \|\Phi_J^{(2)} f\|^2 \leq \|f\|^2.$$

Spectral Geometric Wavelets

Setup - Spectral Representation of the Heat Semigroup

Let Δ be the Laplace-Beltrami operator on a manifold $\mathcal{M}$ with eigenvectors φ_k, $\Delta\varphi_k = \lambda_k\varphi_k$.

$$P_t f(x) = \sum_{k=0}^{\infty} g(\lambda_k)^t \langle f, \varphi_k \rangle \varphi_k, \quad g(\lambda) = e^{-\lambda}$$

Spectral Wavelets

$$\mathcal{W}_J^{(2)} f(x) = \{\Psi_j^{(2)} f(x), \Phi_J^{(2)} f(x)\}_{0 \leq j \leq J},$$

where $\Phi_J^{(1)} = P_{2^J}$ and

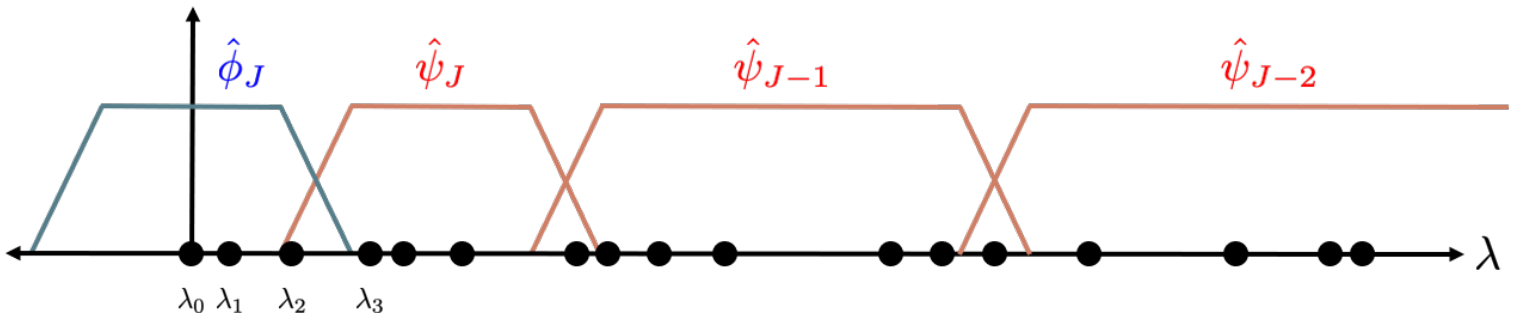
$$\Psi_j^{(1)} = (P_{2^{j+1}} - P_{2^j})^{1/2} = \sum_{k=0}^{\infty} [g(\lambda_k)^{2^{j+1}} - g(\lambda_k)^{2^j}]^{1/2} \langle f, \varphi_k \rangle \varphi_k.$$

Spectral Wavelet's Continued

Theorem: P., Gao, Wolf, Hirn

$\mathcal{W}_J^{(1)}$ is an isometry, i.e.,

$$\sum_j \|\Psi_j^{(1)} f\|^2 + \|\Phi_J^{(1)} f\|^2 = \|f\|^2.$$

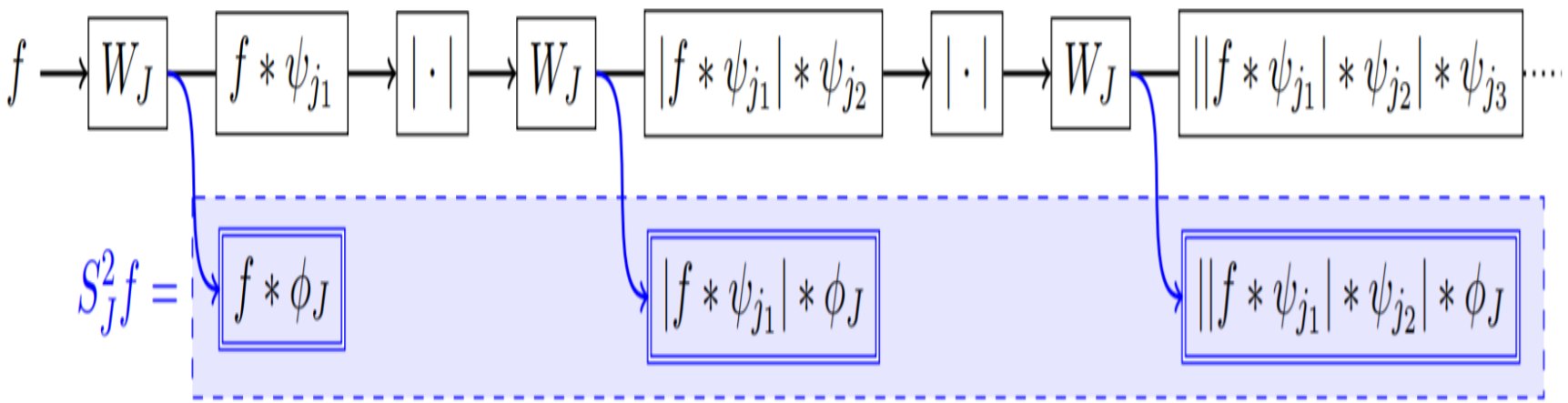


Geometric Scattering

Theorem: (P., Gao, Wolf, Hirn)

The graph and manifold scattering transforms constructed with these wavelets have similar theoretical guarantees to the Euclidean scattering transform:

- Non-expansiveness (Lipschitz continuity on $\mathbf{L}^2$)
- Invariance to manifold isometries of graph permutations
- Stability to perturbations which are close to being isometries / permutations



Trainable Graph Scattering Transform

Scattering Channels

Use many paths of the form $p = (j_1, \ldots, j_m)$:

$$U_p \mathbf{x} := \Psi_{j_m} \sigma(\Psi_{j_{m-1}} \sigma(\ldots \sigma(\Psi_{j_2} \sigma(\Psi_{j_1} \boldsymbol{x})) \ldots).$$

Layer-wise update rule:

$$X_{sct}^{\ell} := \sigma\left(U_p X^{\ell-1} \Theta + B\right).$$

Hybrid Network

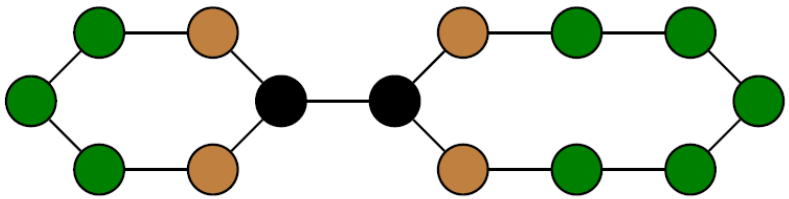
- Min, Wenkel, and Wolf (2021) use both GCN chanels and Scattering channels of each layer.
- GCN channels focus on low-frequency information.
- Scattering Channels retain high-frequency information.

Overcoming oversmoothing

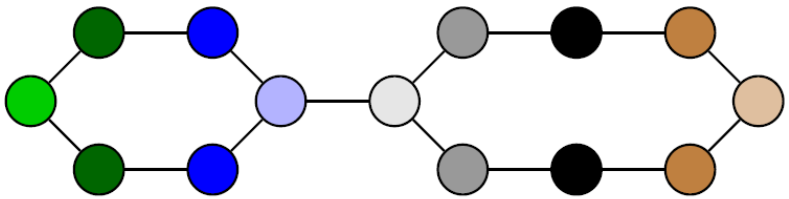
Theorem: (Wenkel, Min, P., Wolf, and Hirn) (forthcoming)

The Hybrid GCN - Scattering network has strictly greater discriminatory power than just GCN

- Introduce a geometric characterization of situations GCN is guaranteed to fail
- Produce a substantial sub-class where scattering will succeed with overwhelming probability.



(a) Graph convolution $(\boldsymbol{g}_\theta \star \boldsymbol{x})$



(b) Geometric scattering $(\Psi_3 \boldsymbol{x})$

Repeated applications of a low-pass filter cause a signal to converge to its projection onto the bottom eigenvector which is either constant or a function of the degree vector.

Graph Scattering Transforms

Untrained Variations

- Zou and Lerman (2020) - Original - spectral wavelets
- Gama, Bruna, and Riberio (2018)- Diffusion wavelets based on T, invariance and stability analysis
- Gao, Wolf, and Hirn (2019)- Diffusion wavelets based on P, statistical moments, graph classification (no theoretical guarantees)
- P. Gao, Wolf, and Hirn (2019)- Invariance and Stability analysis for general diffusion wavelets

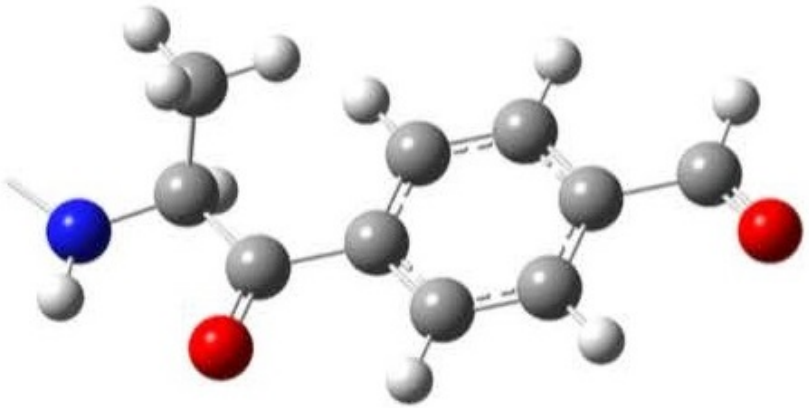
Trained Networks

- Min, Wenkel, and Wolf (2020,2020)- Hybrid network, Attention Mechanism
- Tong et al. (2020) - Learns scales based on data

Graph Generation

Problem:

- Given a dataset of graphs, can you generate a new graph that looks like it was a member of the original dataset
- Motivating Application - Drug Development



Encoding robust representation for graph generation (Zou and Lerman 2019)

- Encoder E = Graph Scattering Transform
- Decoder D = Fully Connected Network
- $D \circ E = Id$
- Generate new graphs by adding noise in latent space

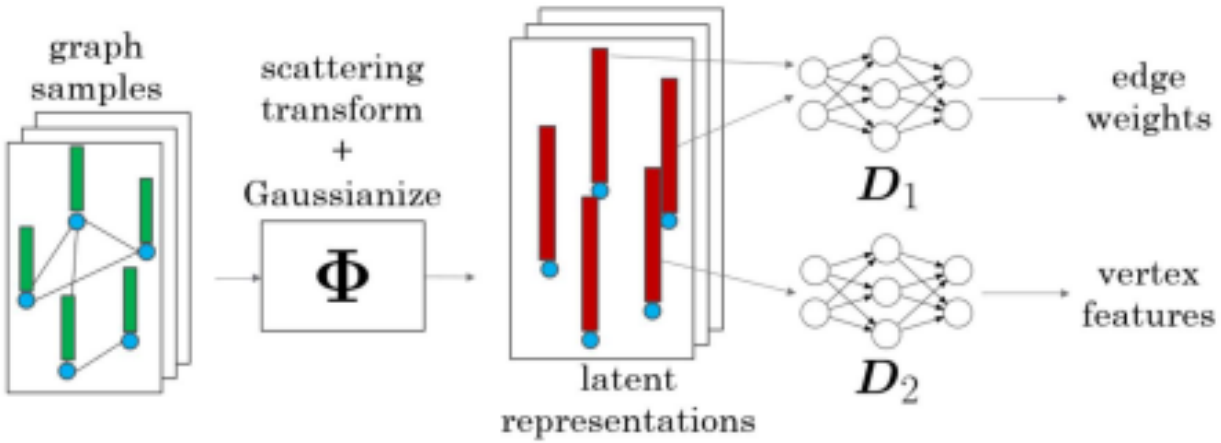


Figure: Scattering Encoder-Decoder Network

Molecular Graph Generation via Geometric Scattering (GRASSY) - Bhaskar, Grady, P., Krishnaswamy

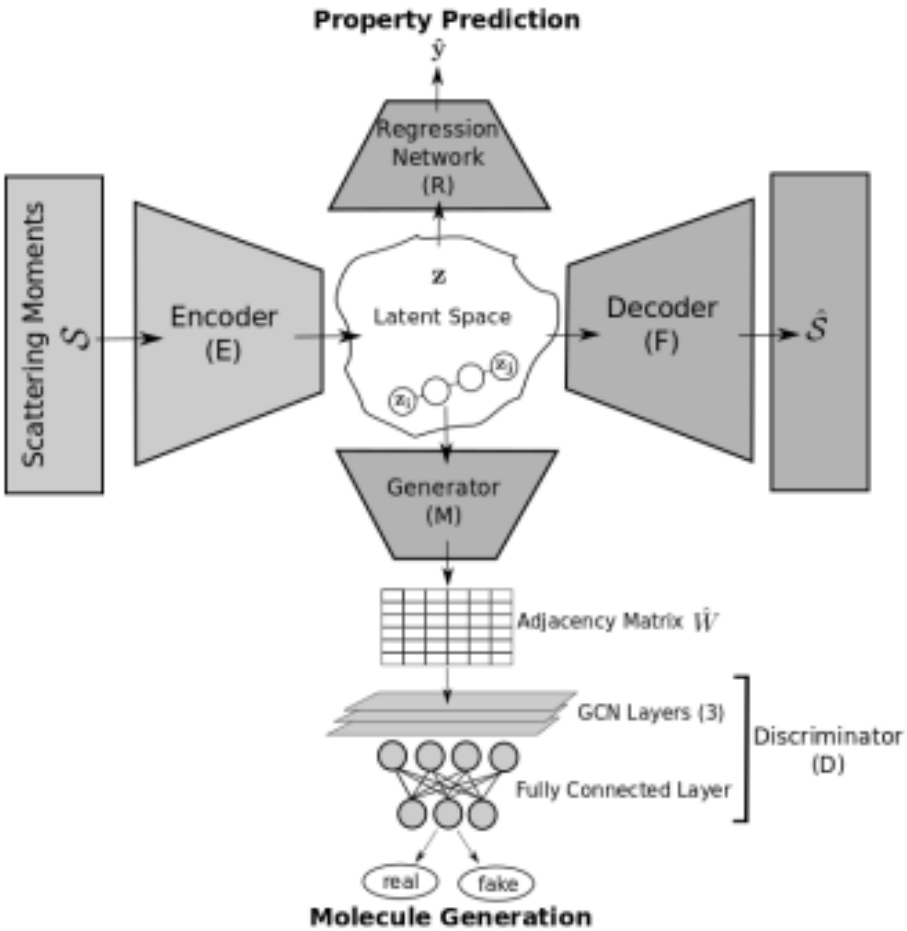


Figure: GRaph Scattering SYnthesis network

GRASSY Explained

- First compute scattering moments to produce a preliminary latent representation.
- Then train a regularized autoencoder on top of the scattering coefficients to produce a compressed latent representation.
- Autoencoder is penalized by a regression network that that aims to predict chemical properties of the molecules.
  - Quantitative Estimate of Drug-likeliness, Molecular Weight, Number of Rings, etc
- Then train a generator / descriminator to generate realistic adjacency matrices.

GRASSY - Bhaskar, Grady, P., Krishnaswamy

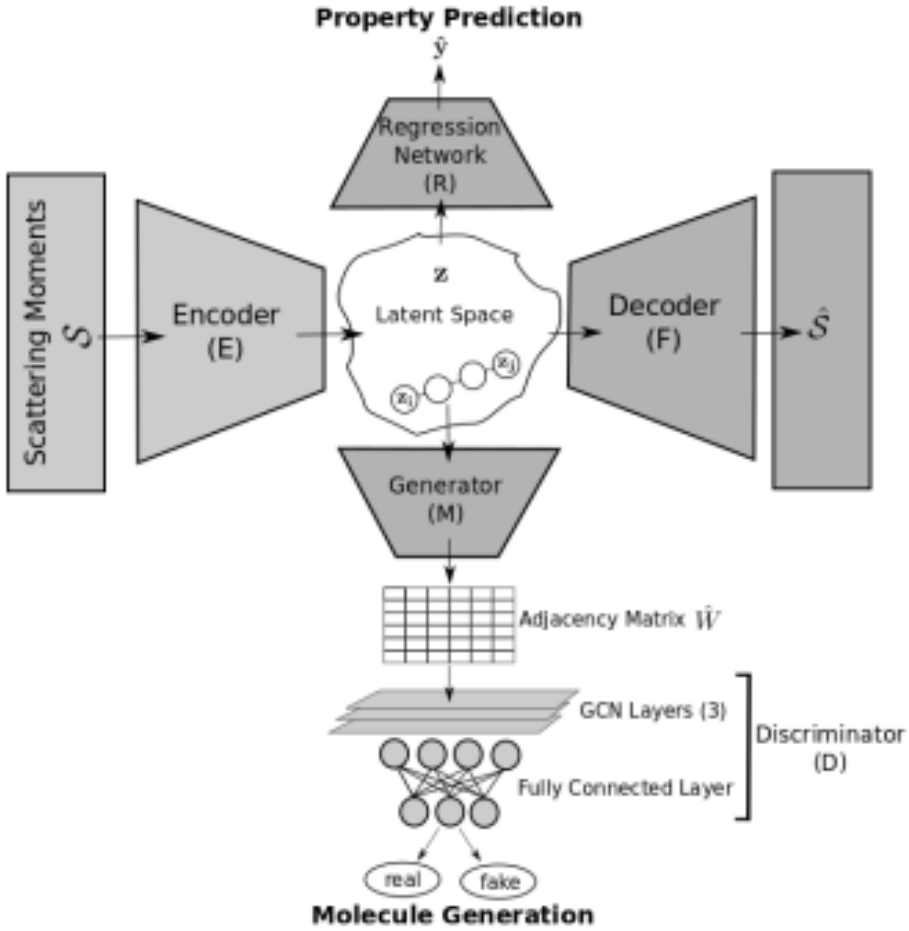


Figure: GRaph Scattering SYnthesis network

Regularized Autoencoder

Loss Functions

- Reconstruction Loss: Want $F \circ E \approx Id$

$$\mathcal{L}_r = \|\mathbf{S} - F(E(\mathbf{S}))\|$$

- Properties Loss: Want to be able to accurately predict Chemical Properties from $E(\mathbf{S})$

$$\mathcal{L}_p = \|p - R(E(\mathbf{S}))\|$$

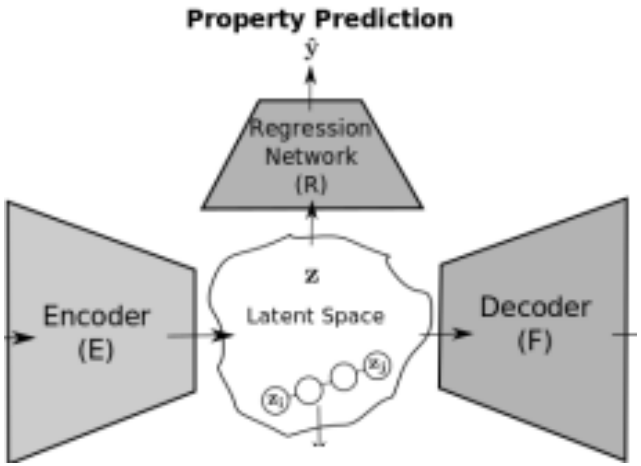


Figure: Regularized AutoEncoder

Molecule Generation

Generative Adversarial Network

- Generator proposes new graphs
- Discriminator decides if they are “real” or “fake”
- Trained in an alternating fashion “against” each other

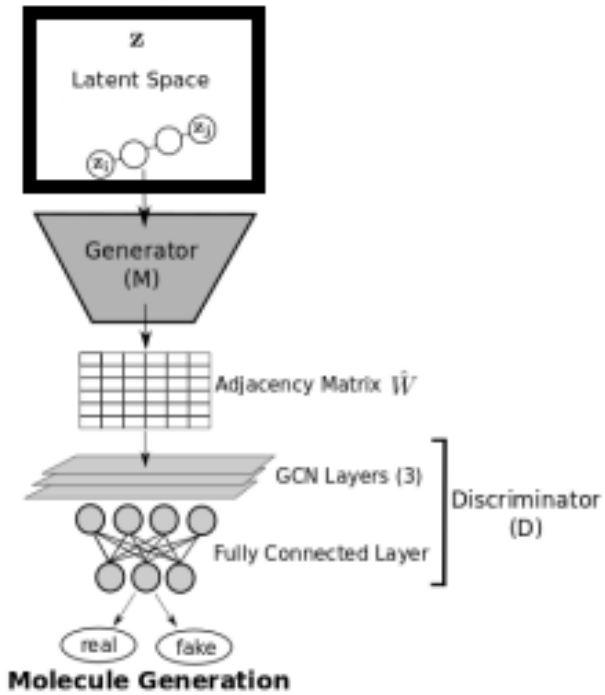


Figure: Generator and Discriminator

Molecule Generation (Cont'd)

Notation

- $z_i = E(\mathbf{S}(G_i))$= Latent representation of $G_i = (V_i, E_i, W_i)$
- $z_{i\to j}(\alpha) = (1-\alpha)z_i + \alpha z_j$
- $\hat{W}_{i,j}(\alpha) = M(z_{i\to j}(\alpha))$ = Matrix generated from $z_{i\to j}$

Losses

- Matrix Loss: $\mathcal{L}_m = \|W_i - \hat{W}_i\| + \|\hat{W}_j - W_j\|$
- Adversarial Loss: $\mathcal{L}_a = -\int \log(D(\hat{W}_{i,j}(\alpha))d\alpha$
- Smoothness Loss: $\mathcal{L}_s \approx \int \|\frac{\partial}{\partial_\alpha} F(z_{i,j}(\alpha))\|^2 d\alpha$

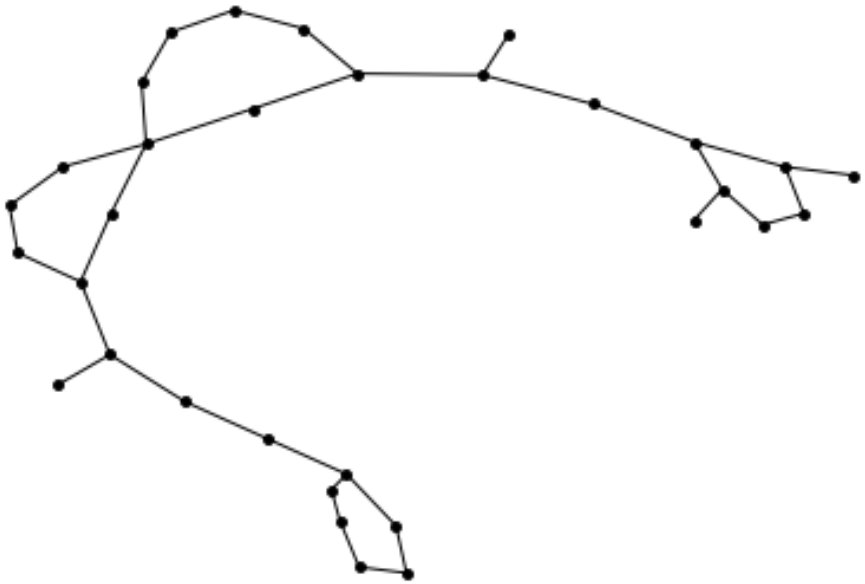
Graphs of Different Sizes

- Pad with zeros

Results

Fraction of graphs generated with molecule-like structure

ZINC	**# Atoms**		**Validity**	**Models**			
Tranche	Min.	Max.	**Threshold**	GRASSY	GSAE	GraphAF	MolGAN ($\lambda = 0$)
BBAB	8	18	5	**0.86**	0.22	0.79	0.32
FBAB	16	27	15	**0.94**	0.17	0.76	0.46
JBCD	28	36	25	**0.73**	0.09	0.54	0.41



Graph Trajectories

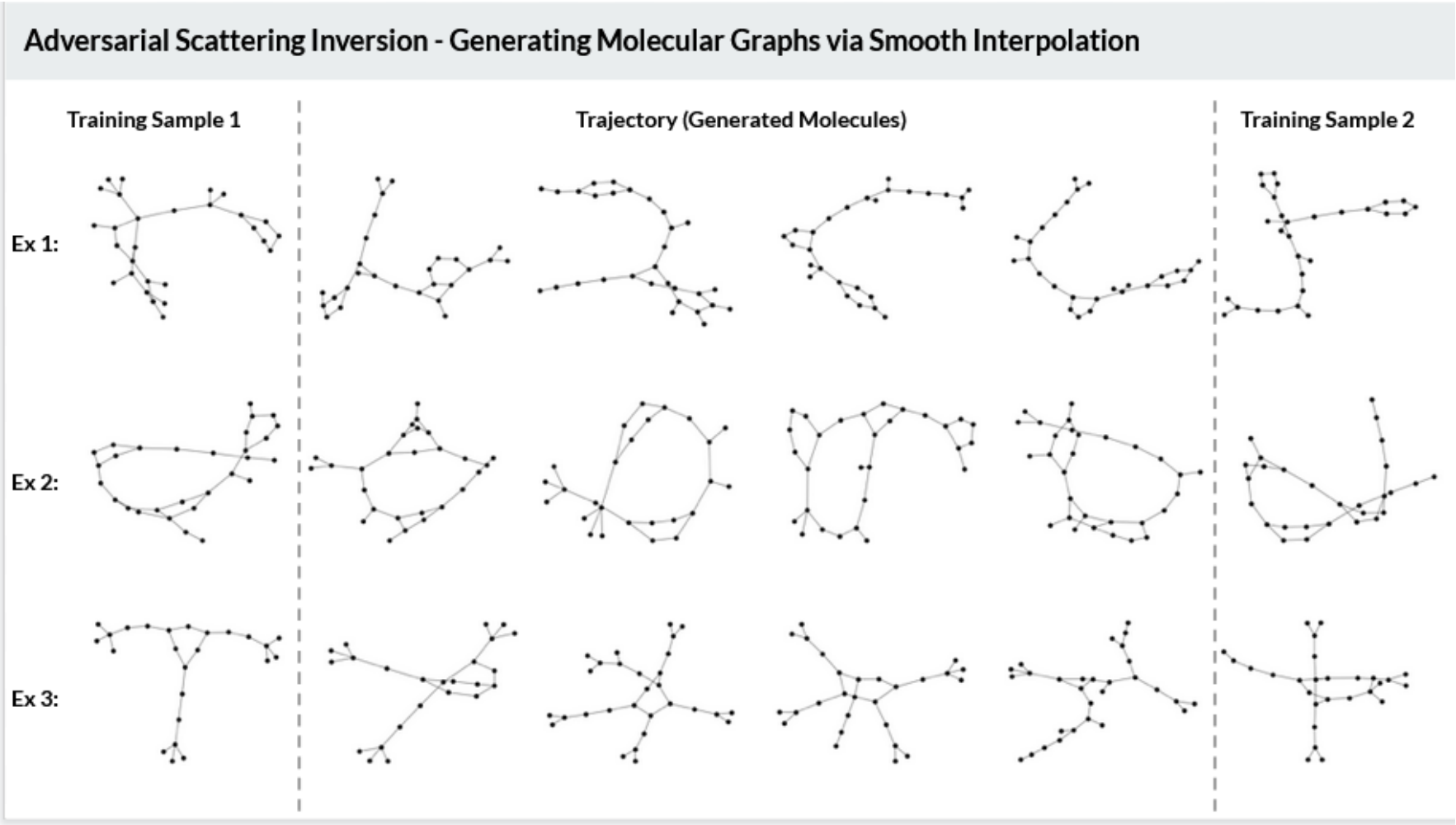


Figure: Trajectories

Latent Space Trajectories

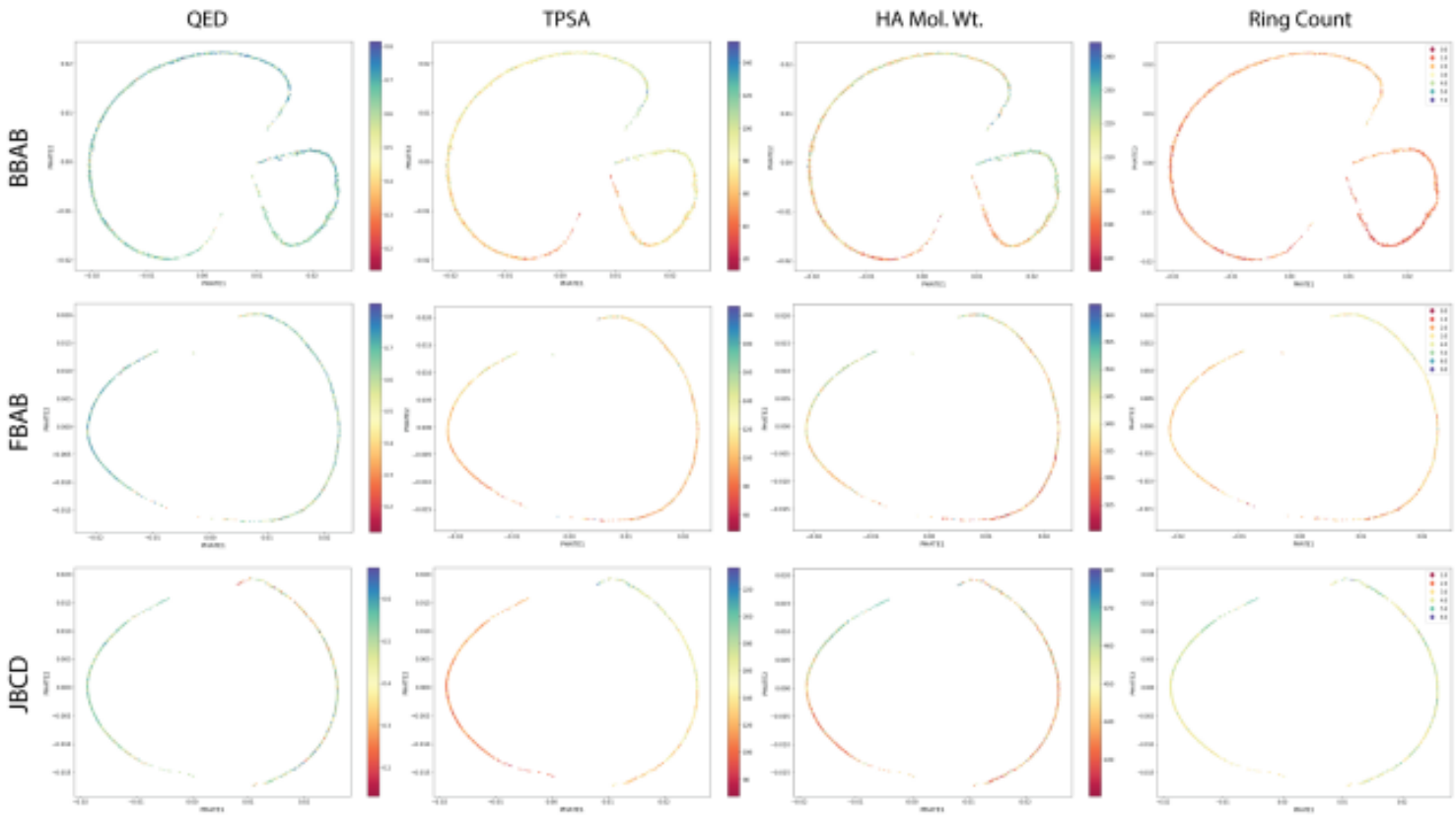


Figure: Latent Representation of molecules from ZINC dataset via the PHATE dimension reduction algorithm

Conclusion

- The Euclidean scattering transform is a model of CNNs.
  - Provable Stability / Invariance Guarantees
  - Designed filters - useful for low-data environments
  - Can be used to synthesize textures
- The graph scattering transform is model of GNNs
  - Similar theoretical guarantees to the Euclidean scattering transform
- The graph scattering transform can be used to synthesize molecules as part of the GRASSY framework
  - Regularized Autoencoder produces compressed representation which respects chemical properties
  - Generator and Discriminator produce new, realistic molecules

THANK YOU!