Spectral ergodicity for explainable deep learning: A simulation approach

Mehmet Süzen

In collaboration with

Cornelius Weber, University of Hamburg
Joan J. Cerdà, University of the Balearic Islands

25 January 2018, MUC

Outline

1. Motivation
2. Circular ensembles
3. Spectral ergodicity
4. Simulations and results
5. Conclusions and outlook

1. Motivation: Success

- State-of-the-art performance in computer vision
  LeCun, Bengio, Hinton, Nature, 521, 436 (2015)

- No explanation so far but success is attributed to

  - *network topology,* multiple processing layers.
  - availability of *large labelled datasets*.
  - leap in *computing capacity*.

- Need a theory or quantification for explanation.

1. Motivation: Scope

- Why do we need to explain?
    - Theoretical understanding: Rethinking generalisation
      Probably Approximately Correct (PAC)
      Vapnik-Chervonenkis (VC)
      Zhang et. al. (2016)
    - Quantify how it works to better tune.
    - No black-box tools.
    - Legal issues in industry.

1. Motivation: Approaches

- Understanding spectral properties are core practice
    - Weight initialization: Dynamical Isometry.
      Pennington et. al. (2017)
    - Singularity in Hessian.
      Sagun et. al. (2016, 2017)
- Information theory: relative compressibility of layers
  Tishby et. al. (2015)

1. Motivation: Our approach

- The concept of *spectral ergodicity*
  Süzen et. al. (2017)

- How to reach to a generic conclusion independent of

  - network architecture (topology).
  - learning algorithm.
  - dataset size and type.
  - training procedure.

- Sample an ensemble of weight matrices.
  Süzen et. al. (2017)

2. Circular ensembles: Introduction

- A realistic simulation of weight matrices
  - Generate ensemble of matrices.
  - Spectral radius reflects stable learning.

- Circular ensembles
  - Spectral radius is close to 1 for *Unitary* ensemble.
  - Numerically stable and mathematically sound generation procedures.

2. Circular ensembles: Parallel Generation

Primary object to generate:
Random Gaussian Hermitian matrix $H \in \mathbb{C}^{N \times N}$

$$H_{ij} = \frac{1}{2}(a_{ij} + Ib_{ij} + a_{ji} - Ib_{ji})$$

$$1 \leq i, j \leq N$$

$a_{ij}, b_{ij}, a_{ji}, b_{ji} \in \mathbb{G}$ i.i.d. Gaussian random numbers

bristol.ensembles.Circular
https://pypi.python.org/pypi/bristol

2. Circular ensembles: Parallel Generation

Circular Unitary Ensemble (CUE) bristol.ensembles.Circular.gen_cue

$$U_{ij} = exp(\gamma_i I) \cdot v_j^i \qquad \gamma_i \in [0, 2\pi]$$

Circular Orthogonal Ensemble (COE) bristol.ensembles.Circular.gen_coe

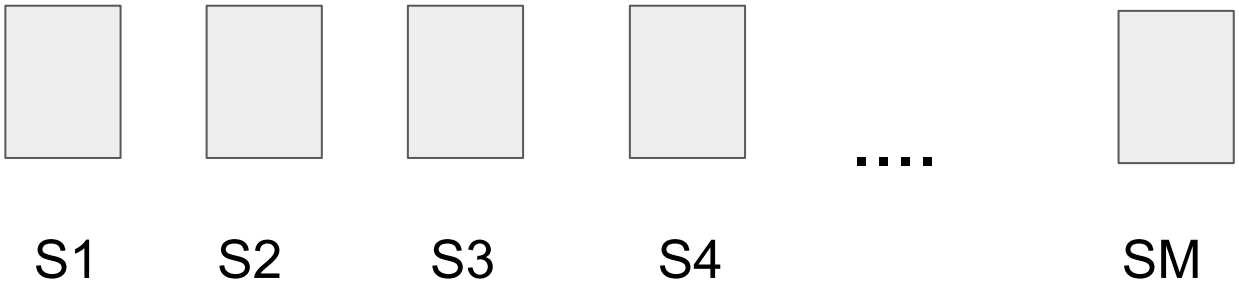
$$O = U^T U$$

Circular Symplectic Ensemble (CSE) bristol.ensembles.Circular.gen_cse

$$S \in \mathbb{C}^{2N \times 2N} \qquad S = (ZU^T Z)U$$

2. Circular ensembles: Random stream chunking

A note on reproducibility against a serial implementation



Given ensemble of size M, retain seeds vector (S1, S2, S3, S4, …, SM), so each processor/chunk gets the same random stream chunking regardless of parallel or serial execution.

bristol.ensembles.Circular.eigen_circular_ensemble

2. Circular ensembles: Interpretation

- Given network architecture:
    - Ensemble of weight matrices for different layers in feed forward networks.
    - Ensemble of weight matrices for entire network in non-sparse recurrent networks.

- Sample ensemble of circular matrices M times,
  number of hidden layers with different sizes N,
  depth in the given layer.

3. Spectral Ergodicity: Definition

- Sample ensemble of circular matrices `M` times, with different sizes `N` and given bin size `b`.
  - Time-averaged eigenvalue-spectra: Compute eigenvalue spectra for `j-th` realization

    $\rho_j(b_k)$
  - Ensemble-averaged eigenvalue-spectra: Compute average spectra over M realisation.

    $$\bar{\rho}(b_k) = \frac{1}{M} \sum_{j=1}^{M} \rho_j(b_k)$$

3. Spectral Ergodicity: Definition

- Approach to spectral ergodicity for given weight matrix size, corresponds to depth of a given layer.

$$\Omega_k^N \equiv \Omega^N(b_k) = \frac{1}{M \cdot N} \sum_{j=1}^{M} \left[\rho_j(b_k) - \bar{\rho}(b_k)\right]^2$$

- Inspired from physical definition of ergodicity, Süzen (2014) recall diffusion coefficient over time.

3. Spectral Ergodicity: Definition

- How far would *spectral ergodicity* drift with an additional depth? A *distance metric*

$$D_{se}(N_a, N_b) = D_{KL}(\Omega^{N_a} || \Omega^{N_b}) + D_{KL}(\Omega^{N_b} || \Omega^{N_a})$$

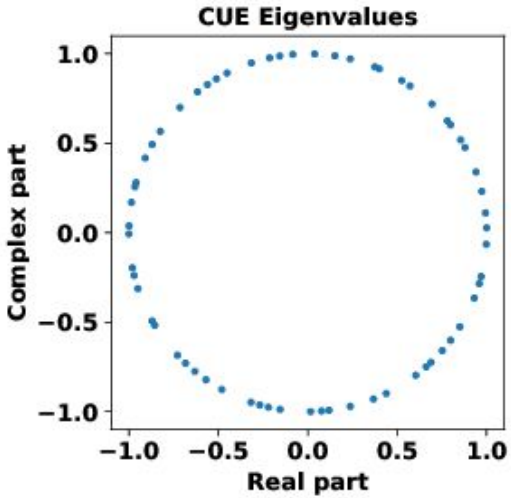
$$D_{KL}(\Omega^{N_a} || \Omega^{N_b}) = \sum_{k=1}^{K} \Omega_k^{N_a} \log_2(\Omega_k^{N_a} / \Omega_k^{N_b})$$

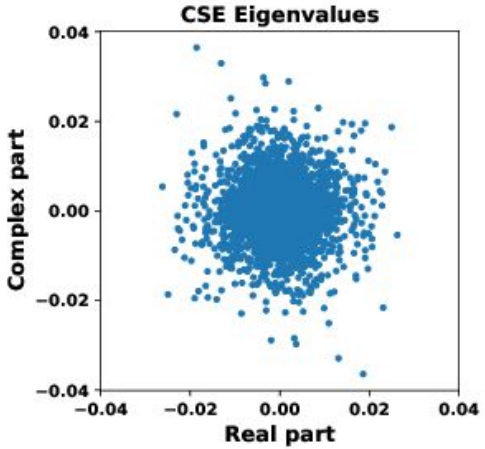
4. Simulations and Results: Summary

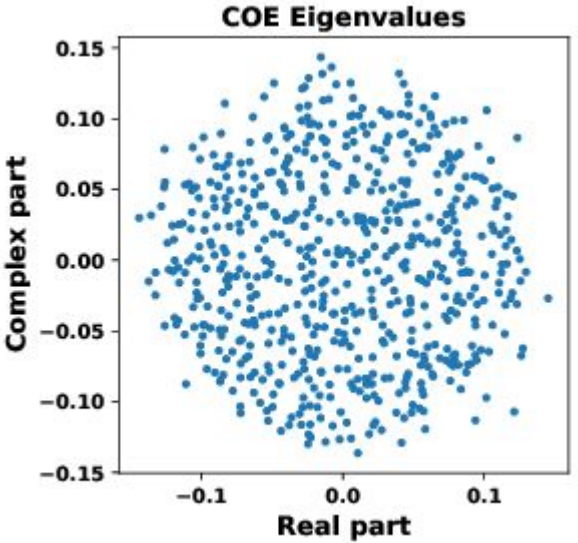
- Ensemble size, M=40, recall, number of hidden layers.
- Weight matrix sizes, N=64, 128, 256, 512, 768, 1024, recall, layer depth.
- Generate all matrices.
- Generate eigenvalues, data available publicly. https://zenodo.org/record/822411#.WmX1AHWnHqM
- Compute the distance metric for increasing depth transitions.
  - 64-128, 128-256, 256-512, 512-768, 768-1024
  - Jupyter Notebook, available publicly. https://github.com/msuzen/bristol/blob/v0.2.2/works/spectralErgodicity/02_ergodicity_random_matrix.ipynb

4. Simulations and Results: Eigenvalues

- Sample of eigenvalues for different ensembles.

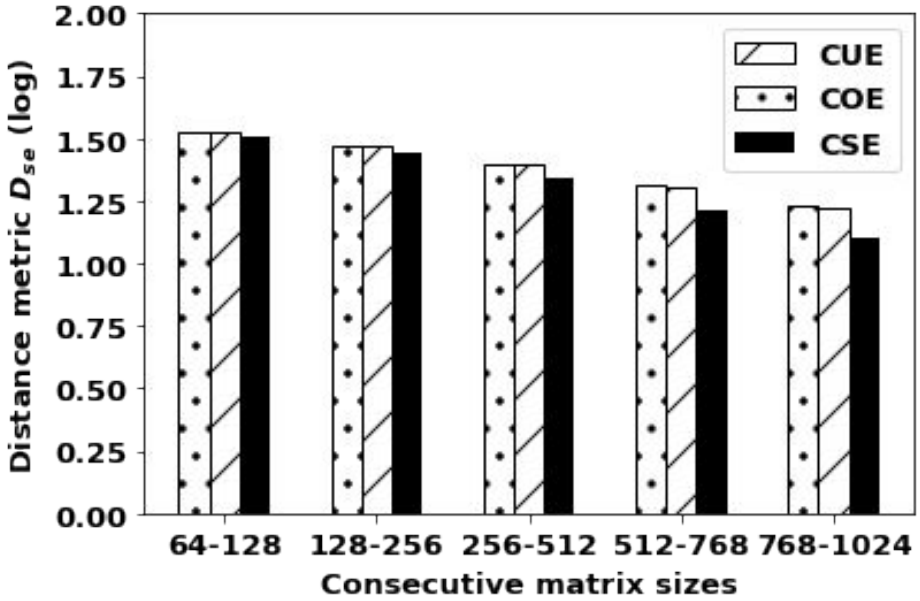






4. Simulations and Results: Spectral ergodicity

- Distance metric.



5. Conclusions and outlook

- We attribute the success of deep learning to spectral ergodicity of a given topology and learning algorithm using surrogate random matrices.
- Interpretation of ensemble can be different layers or entire network.
- The work can be extended to synaptic matrices in the context of spiking neurons.

Thank you
Q & A

Spectral Ergodicity in Deep Learning Architectures via Surrogate Random Matrices
Mehmet Süzen, Cornelius Weber, Joan J. Cerdà
arXiv preprint arXiv:1704.08303 (2017)
dataset zenodo doi: 10.5281/zenodo.822411
code zenodo doi: 10.5281/zenodo.579642

Python package:
https://pypi.python.org/pypi/bristol