Free Random Projection for In-Context Reinforcement Learning

Tomohiro Hayase

arXiv:2504.06983 **[cs.LG]**
Code available at https://github.com/ThayaFluss/frp_rl
Joint work with Benoit Collins, Nakamasa Inoue.

[Non-Main Topic]
Gaussian Equivalence for Self-Attention: Spectral Analysis of Attention Matrix
arXiv:2510.06685 **[stat.ML]**
Joint work with Benoit Collins, Ryo Karakida.

Self-Introduction

- Tomohiro Hayase – a researcher spacializing in mathematical science and machine learning.
- Research Interest: Capturing seemingly intractable phenomena like **randomness and infinite-dimensional** structures using **finite-dimensional concepts** for computation.

Research Axes and Related Fields

- Symmetry & Hierarchy
  - Symmetric/Hierarchical structures in problems.
- Free Probability & Random Matrices
  - Asymptotic Freeness and Applications.
- Infinite Width Neural Networks & Kernel
  - Deep Neural Network in the inifinite-width-limit.
  - arXiv:2510.06685 : **Spectral Analysis of Attention Matrix**
- Generalization & Implicit Bias
  - ML model generalization and implicit inductive bias.

Today's talk, FRP(arXiv:2504.06983), is a combination of All axes of the above !

Goal of Talk: New Application of Free Probability & Random Matrices to ML, in particular, Reinforcement Learning.

Core: Asymptotic Freeness（ O(d), L2-ver.）

Let $U_1, \ldots, U_n \sim \mathrm{Unif}(\mathrm{O}(d))$, i.i.d., $\lambda : \mathbb{F}_n \to \mathrm{O}(d)$; $\lambda(a_i) = U_i$ $(i = 1, 2, \ldots, n)$. Then

$$\lim_{d\to\infty} \mathbb{E}[\langle \lambda(v), \lambda(w) \rangle_{\mathrm{O}(d)}] = \langle v, w \rangle_{\mathbb{F}_n} \quad \text{for any } v, w \in \mathbb{F}_n,$$

where $\langle U, V \rangle_{\mathrm{O}(d)} = d^{-1}\mathrm{Tr}U^{\mathsf{T}}V$, and $\langle v, w \rangle_{\mathbb{F}_n} = 1$ if $v = w$ and 0 otherwise.

The word problem for Haar orthogonal matrices (i.e., determining when a word in the generators equals the identity) is asymptotically equivalent to that for free groups.

1. Introduction
2. **Background : Generalization and Implicit Bias**
3. Free Group and Random Matrices
4. Free Random Projection
5. Future Work

Background & Problem Awareness

- Need for Generalization

  ML models are expected to perform well even in **new situations** different from the training environment (i.e., to generalize)

- Inductive Bias

  We often assume some common structure between training and test environments and **build that into the model** as an inductive bias

- Implicit Bias (or Implicit Regularization)

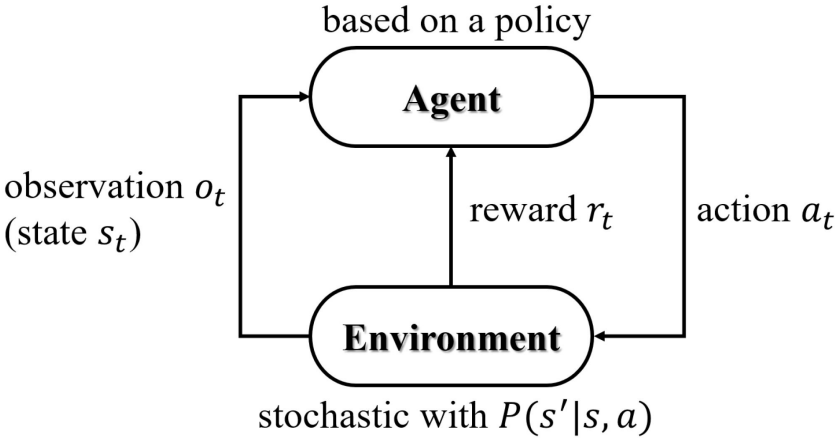
  Even model with weak explicit structure (like large DNNs) can generalize well. This suggest the presence of an implicit bias (**stemming from the training algorithm** rather than model architecture)

  - [G. Backman, NeurIPS2023]Scaling MLP: a talk of inductive bias
  - [R. Karakida+, ICML2023]Understanding Gradient Regularization in Deep Learning: Efficient Finite-Difference Computation and Implicit Bias

Reinforcement Learning Basics

Agent & Enviroment

- Agent observes a state, takes an action, and receive a reward.
- Gols is to learn a policy that maximizes the cumulative reward through interactions.



Markov Decision Process

MDP: (S, A, R, P, \gamma)

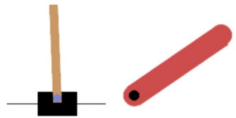
Partially Observable MDP

Agent can only observe a part of states.

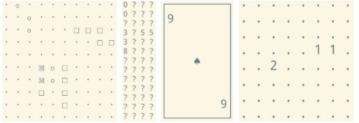
Generalization in Meta RL

Multi-Environment Setting

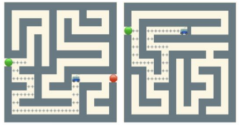
- A major challenge in RL is generalization across multiple environments/tasks.
- An agent trained on certain environments should adapt and perform well in unseen environments.



(a) Stateless Cartpole and Stateless Pendulum



(b) Battleship, Concentration, Higher Lower and Mine Sweeper



(c) Labyrinth Escape and Explore

Meta RL

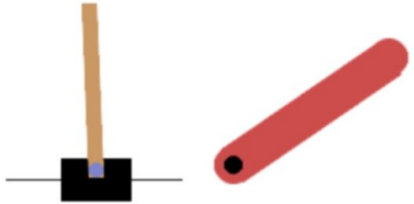
- Agent learns commonalities across tasks
- so that it can quickly adapt to new task

In-Context RL

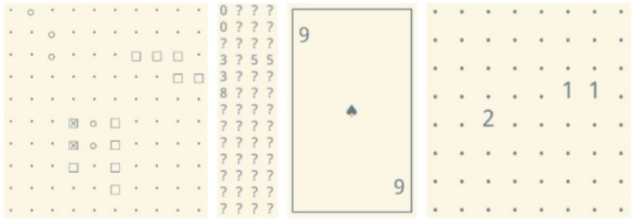
- Meta-RL only changing hidden states in models

Meta & In-Context Reinforcement Learning

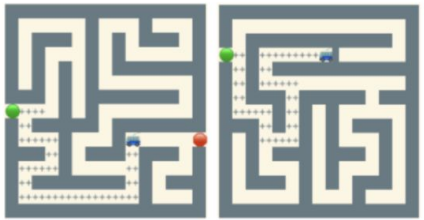
e.g. POPGYM



(a) Stateless Cartpole and Stateless Pendulum

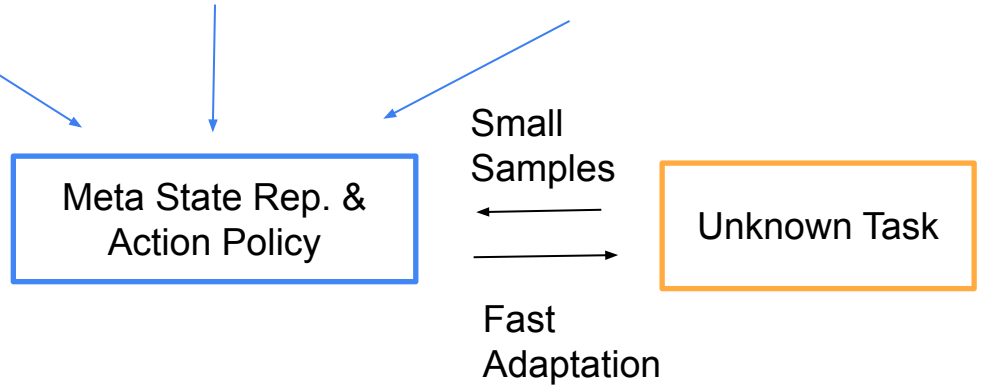


(b) Battleship, Concentration, Higher Lower and Mine Sweeper



(c) Labyrinth Escape and Explore

e.g. Morad+,ICLR2023,
Lu+, NeurIPS2023.



In-Context RL

- A from of meta-RL where the agent **adapts without updating its parameters**, by inferring the task from *the context* (the sequence of observed states, actions, rewards.)

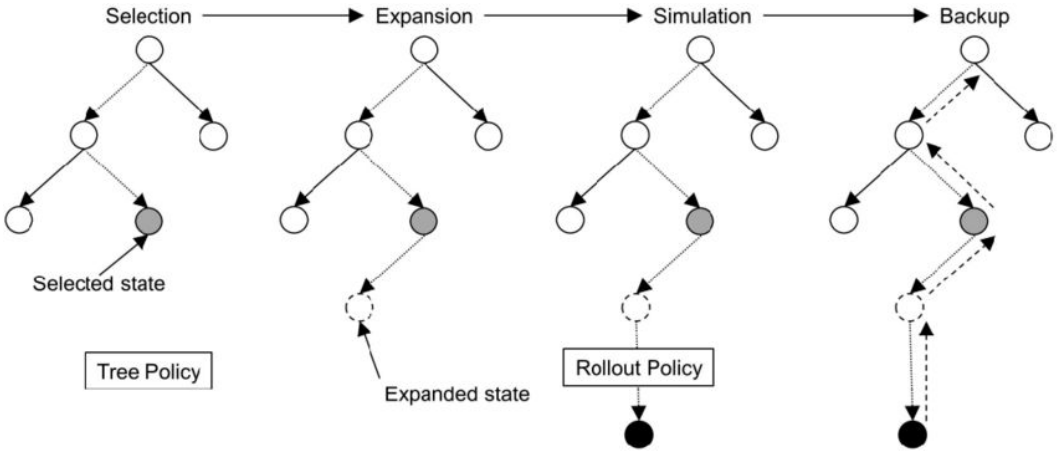
  Within the decision-time context windows, the agent adjusts its policy **on the fly** to handle a new environment.

- Common Approach

  When dealing with differing input formats across tasks, a **random projection** is often used to embed each environment int a unified dimension. (Applying Uniformly distributed Isometry, a kind of Haar Orthogonal Random Matrix)

Need for Hierarchical Bias (1/3)

- Hierarchical Tasks: Many tasks in RL exhibit a **tree-like** or hierarchical structure (e.g. tree of future state transition branches out exponentially like a tree)
- Algorithm with Tree: MCTS: (e.g. AlphaZero, AlphaGo)



Need for Hierarchical Bias (2/3)

- Hyperbolic Geometry

Recent research shows the hyperbolic spaces can capture hierarchical relationships more effectively than Euclid spaces. (e.g. using hyperbolic embeddings to hierarchical structures)

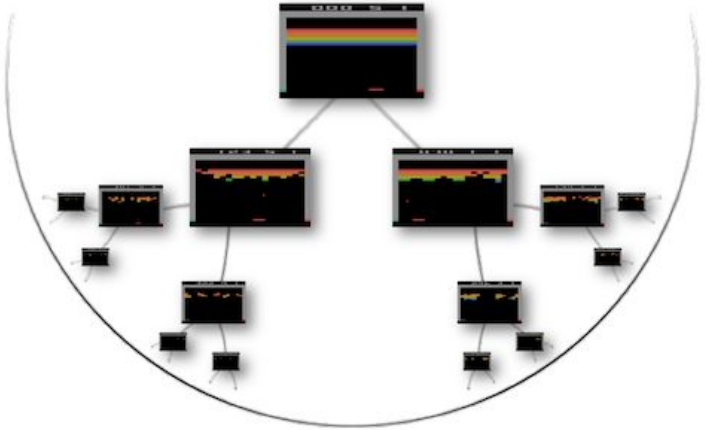


Figure 1: Hierarchical relationship between states in *breakout*, visualized in hyperbolic space.

Figure from Cetin+ ICLR2023.

Need for Hierarchical Bias (3/3)

- Hyperbolic Geometry
- However, most such approaches rely on **specialized model architectures** or loss functions, meaning the hierarchical bias must be explicitly built in.
- *Our Hypothesis*: It would be more flexible if **the learning algorithm itself induces a hierarchical bias naturally**, without requiring complicated changes to the model architecture. This work aims to realize such an algorithmic bias.

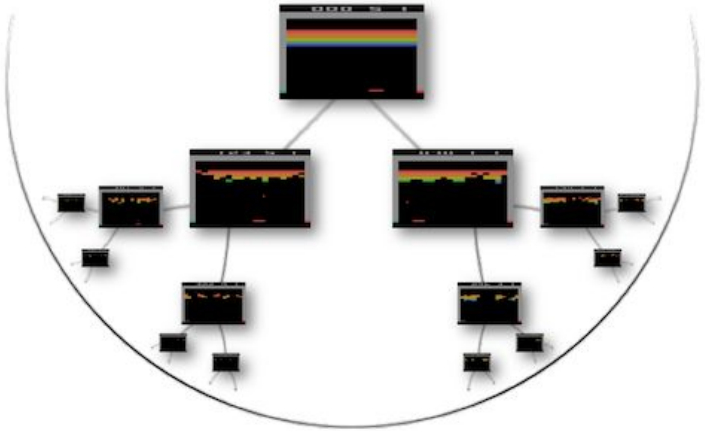
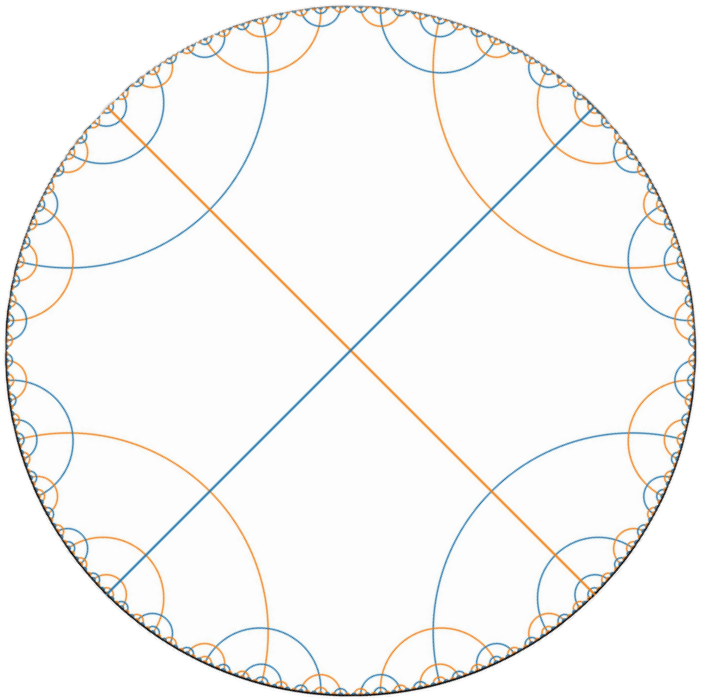


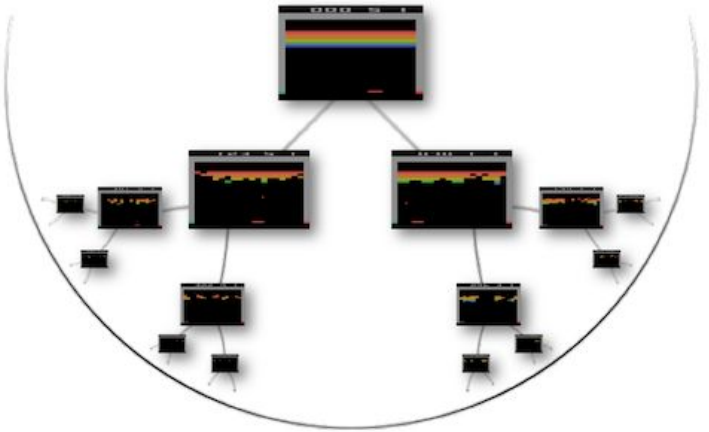
Figure 1: Hierarchical relationship between states in *breakout*, visualized in hyperbolic space.

Figure from Cetin+ ICLR2023.

1. Introduction
2. Background : Generalization and Implicit Bias
3. **Free Group and Random Matrices**
4. Free Random Projection
5. Future Work

Use Free Group for Hyperbolic/Hierarchical Properties





Free Monoid and Free Group

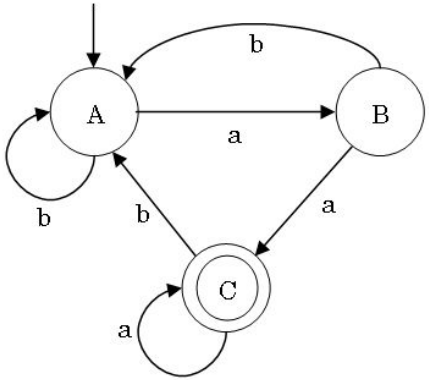
Free Monoid: Take an alphabet (set of generators); consider the set of all words formed from these symbols. This set, equipped with **word concatenation** as the multiplication, is a free monoid.

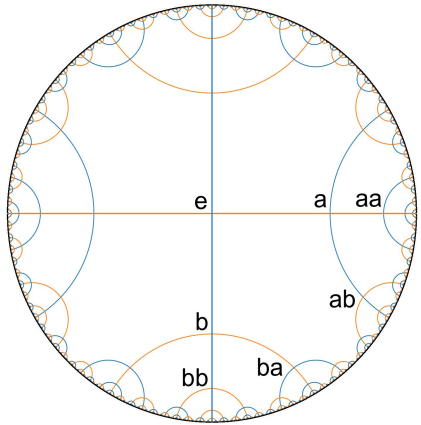
w=ab, v=aaa => wv = abaaa.

Traditionally, the free monoid is often used in CS, e.g. in **formal language & automaton**.

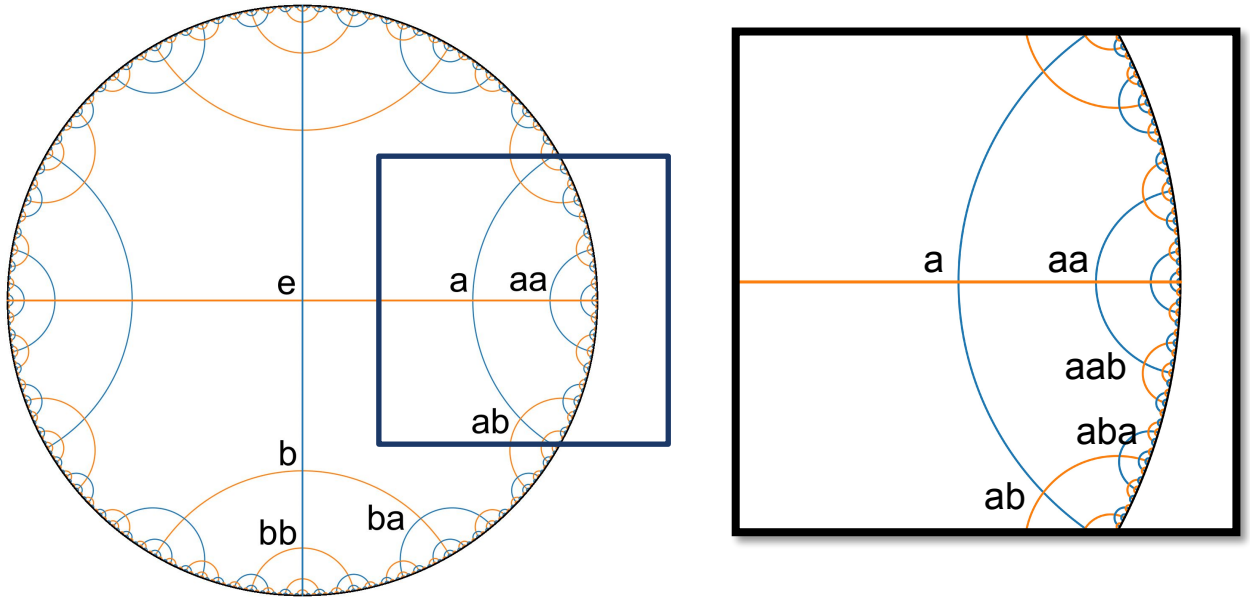
Free Group: Start with a free monoid and then introduce a formal inverse for each generator. The resulting structure is a free group.

It's **Cayley Graph** is a **Tree** and a **Gromov Hyperbolic space.**





Free group has infinite hierarchy



Core: Asymptotic Freeness（ O(d), L2-ver.）

Let $U_1, \ldots, U_n \sim \mathrm{Unif}(\mathrm{O}(d))$, i.i.d., $\lambda : \mathbb{F}_n \to \mathrm{O}(d)$; $\lambda(a_i) = U_i$ $(i = 1, 2, \ldots, n)$. Then

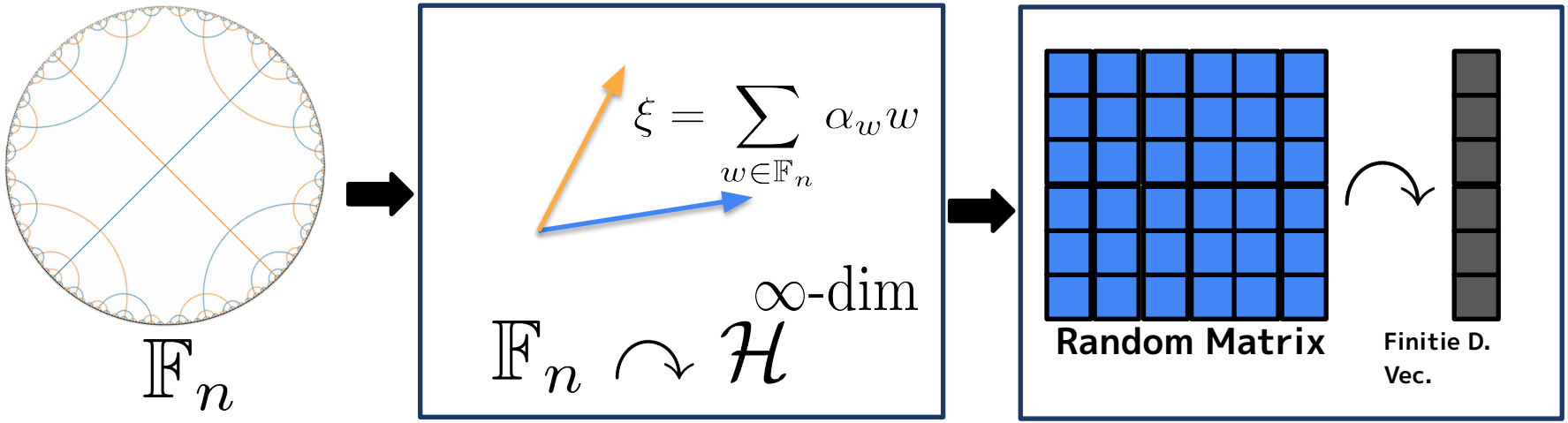
$$\lim_{d\to\infty} \mathbb{E}[\langle \lambda(v), \lambda(w) \rangle_{\mathrm{O}(d)}] = \langle v, w \rangle_{\mathbb{F}_n} \quad \text{for any } v, w \in \mathbb{F}_n,$$

where $\langle U, V \rangle_{\mathrm{O}(d)} = d^{-1}\mathrm{Tr}U^{\mathsf{T}}V$, and $\langle v, w \rangle_{\mathbb{F}_n} = 1$ if $v = w$ and 0 otherwise.

The word problem for Haar orthogonal matrices (i.e., determining when a word in the generators equals the identity) is asymptotically equivalent to that for free groups.

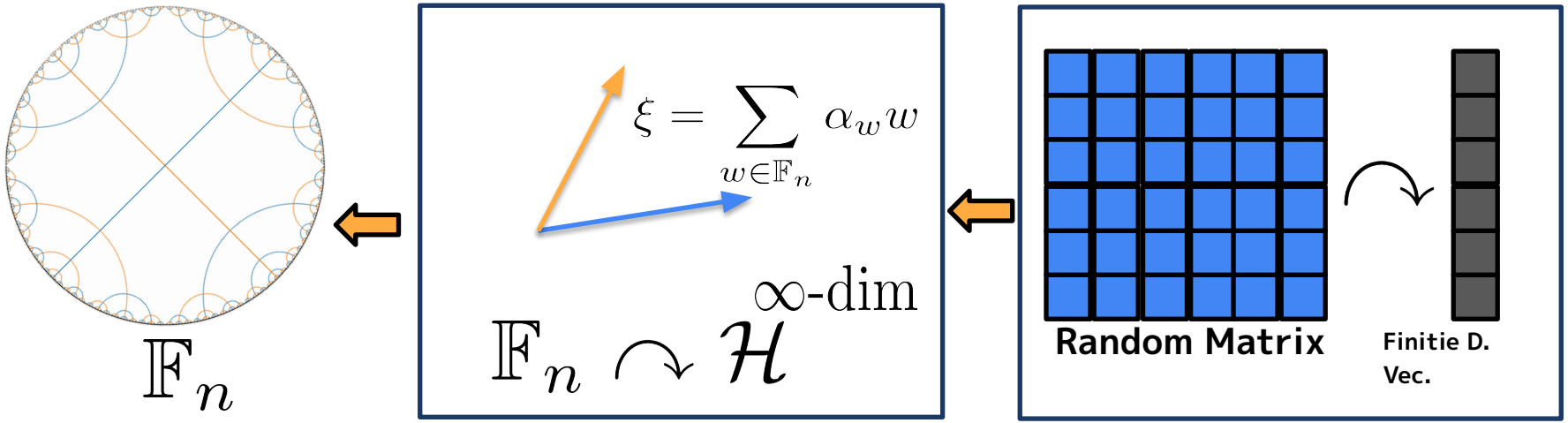
Key Point: Linearization and Finite Dimensional Approx.

- Group it self is difficult to tree： Vectorize and insert similarity（Group Hilbert Space）.
- Inifinite dimensonal matrices can be un-computable： Apprxomate it with finite dimensional random matrices.



Application of “Free” was inverse direction

- Random matrices are used in deep learning & meachine learning researche.
  Compute Deep Net’s MFT/NNGP/NTK/DI（or related quantities）with
  approximating random matrices with Free groups



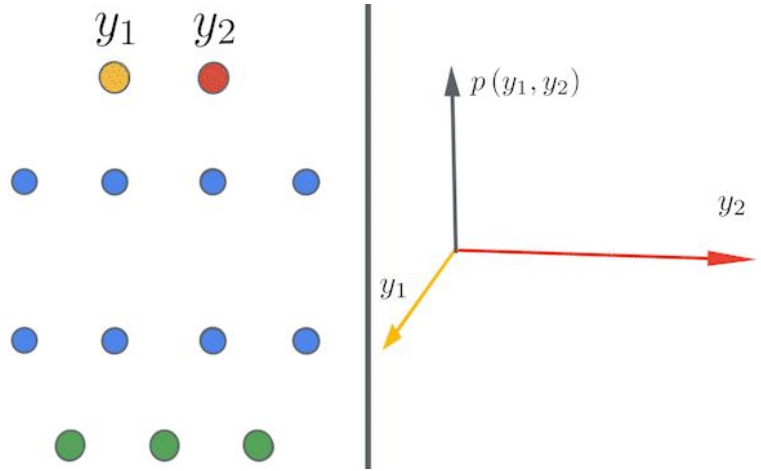
Example of the inverse direction

Inifinite Wide limit: Hidden vectors are Guasiaan

Output： Neural Network Gaussian Process (NNGP)

Gradient： Neural Tangent Kernel (NTK)

- Spectral Analysis of Forward/Backward signal ~ using freeness, S-transform, free convolutions.



21

1. Introduction
2. Background : Generalization and Implicit Bias
3. Return to Math : Free Group and Random Matrices
4. **Free Random Projection**
5. Future Work

Total 41 pages

Free Random Projection

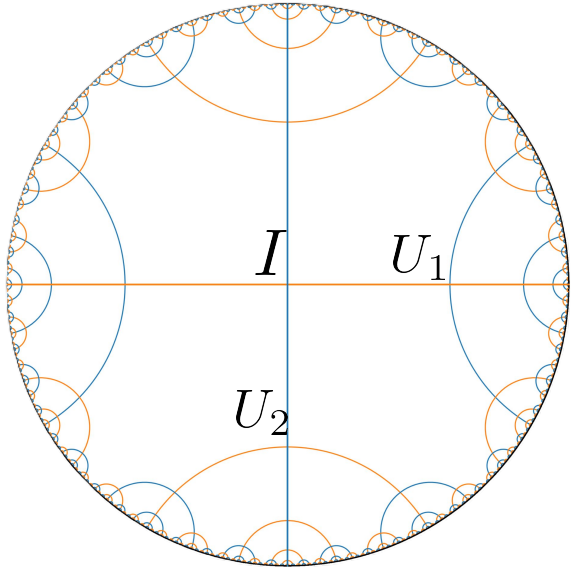
First, sample d by d Haar Orthogonal matrices, corresponding to Free group generators:

$$U_1, U_2, \ldots, U_n.$$

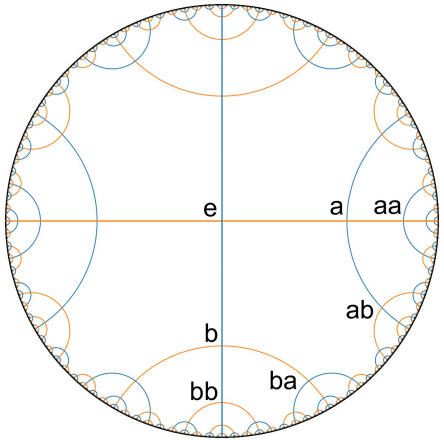
Compoue the projection matrix corresponding

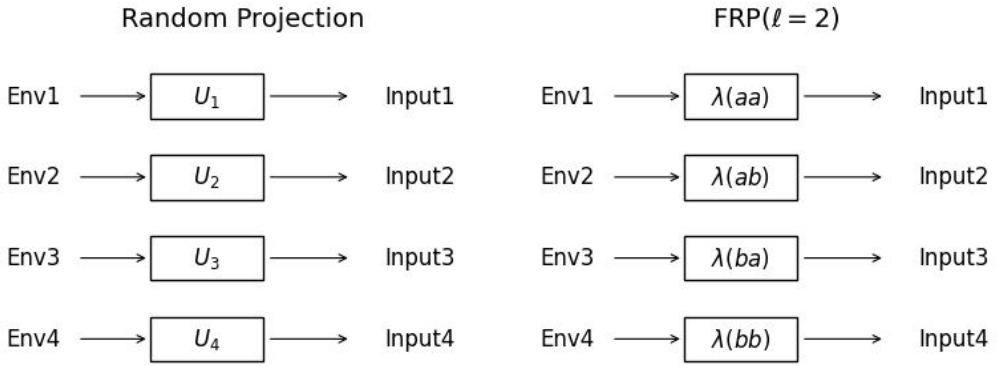
to a sampled word:

$$i_1, i_2, \ldots, i_\ell \in [n].$$
$$A = U_{i_1} U_{i_2} \cdots U_{i_\ell}$$



Free Random Projection





1. During Training, periodically re-sample base random matrices.
2. For each environment, sample a word from a fixed word distribution.

$$U_1, U_2, \ldots, U_n.$$
$$A = U_{i_1} U_{i_2} \cdots U_{i_\ell}$$

Word Distribution

- ℓ: The length of words.
- n: The number of generators.
- n_e: The number of environments.

We define the following collection of words:

$$\Lambda_n^\ell := \{a_{i_1} \dots a_{i_\ell} \in \mathbb{F}_n \mid i_1, \dots, i_\ell \in [n]\} = \langle a_1, \dots, a_n \rangle^\ell. \tag{1}$$

We define the word distribution as

$$\rho_{\mathbb{F}} := \mathrm{Unif}(\Lambda_n^\ell) \tag{2}$$

and sample words $\{w_1, \dots, w_{n_e}\}$ independently from $\rho_{\mathbb{F}}$ for the n_e parallel environments.

In our analysis, we fix the total number of possible words $n_w > n_e$ and compare various pairs of (n, ℓ) that satisfy $|\Lambda_n^\ell| = n_w$ to ensure a fair comparison.

Meta RL environment step with FRP

Algorithm 1 Meta RL environment step with FRP

Require: Distribution of environments $\rho_{\mathcal{E}}$, Agent action a and Environment termination 1^{done}
Require: Distribution of words $\rho_{\mathbb{F}}$, Matrix representation $\lambda : \mathbb{F}_n \to \mathrm{O}(d)$.

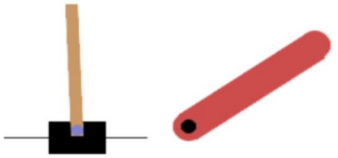
function StepEnvironment(a, 1^{done})
  if the environment terminated (1^{done}) **then**
    Sample random environment $E \sim \rho_{\mathcal{E}}$
    Sample random word $w \sim \rho_{\mathbb{F}_n}$
    Initialize random observation projection matrix $M_o = \sigma_w T_2 \lambda(w) T_1^E$
    Initialize random action projection matrix M_a
    Reset E to receive an initial observation ξ
    Apply the random observation projection matrix to the observation $\xi' = M_o \xi$
    Initialize $r = 0$ and $1^{\text{done}} = 0$
    return $\xi', r, 1^{\text{done}}$
  else
    Apply the projection matrix $a' = M_a a$
    Step E using a' to receive the next observation ξ, reward r, and done signal 1^{done}
    Apply the projection matrix $\xi' = M_o \xi$
    return $\xi', r, 1^{\text{done}}$
  end if
end function

Toy Experiment: Effect of Hierarchy

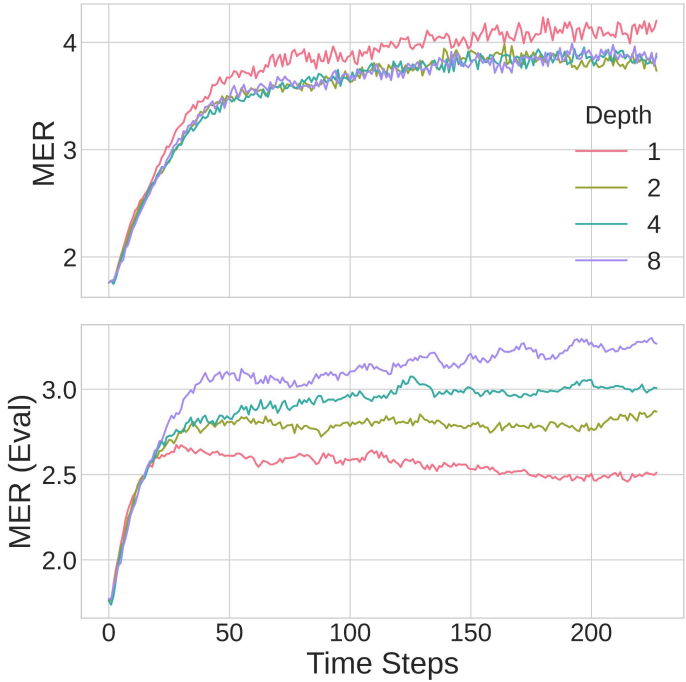
In-Context & Meta RL experiment with Cartpole(Only position is observable) + Resettable S5(an RNN).

We compared learning performance using FRP with different depths (word lengths). (Depth = 1 corresponds to the conventional RP.)
The depth represents the length of the free group word; larger depth means the projection carries more hierarchical structure.

We found that deeper FRP (larger word length) leads to better generalization!



(a) Stateless Cartpole and Stateless Pendulum



Experiment: Multi-Task Comparison

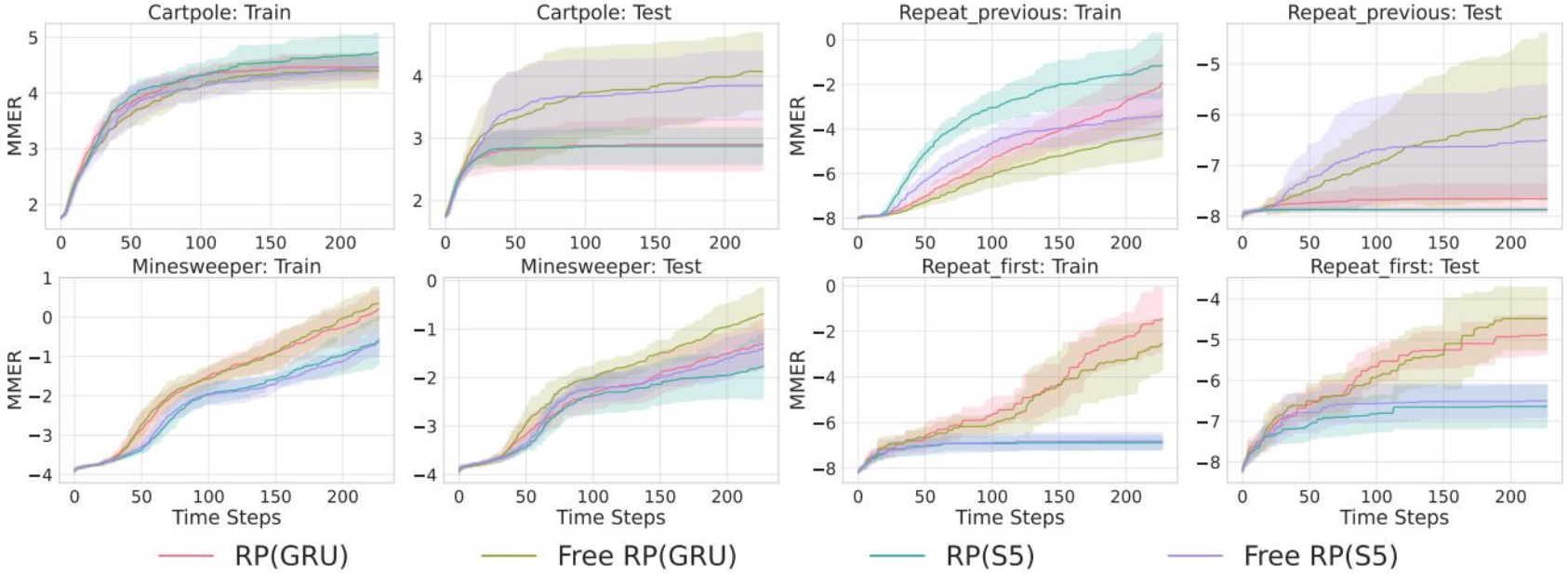


Figure 2: Performance of FRP vs. standard RP on four environments – Stateless Cartpole, Repeat Previous, Mine Sweeper, and Repeat First – shown in the top-left, top-right, bottom-left, and bottom-right subplots, respectively. We use ℓ^* in Table 2 for FRP. Each subplot plots Train MMER and ICL-Test MMER. Shaded regions indicate standard error across 10 random seeds.

Quantitative Results

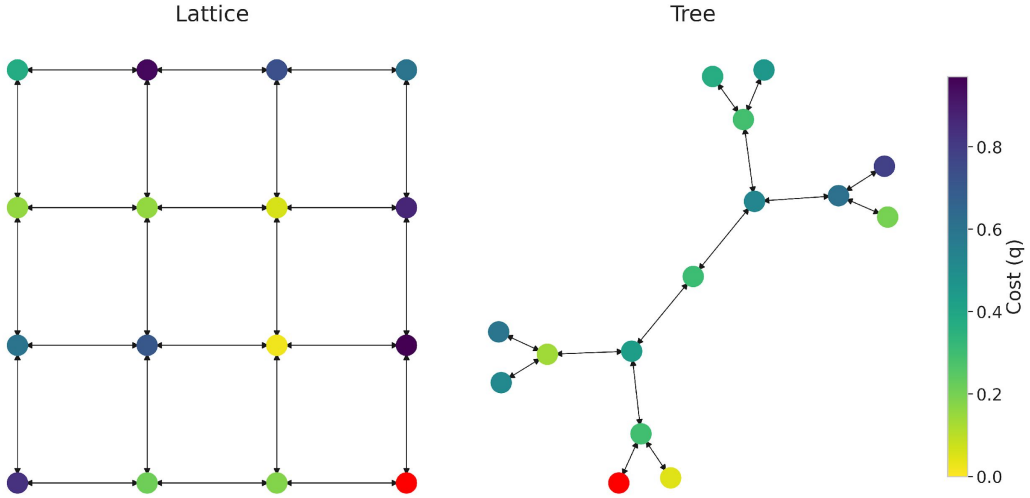
The **FRP + GRU** combination achieved the best performance across all tasks. It outperformed other comparison models and consistently achieved higher Test-MMER than the standard random projection baseline.

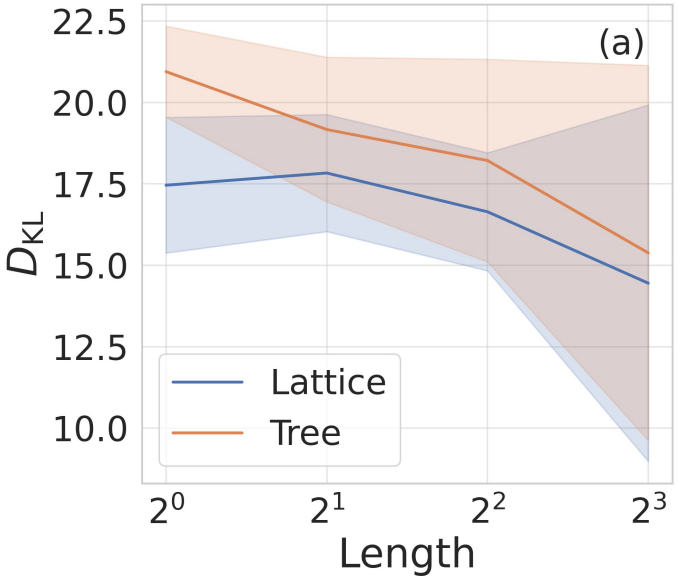
Table 1: Test performance comparison across POPGym environments; Stateless Cartpole, Higher Lower, Mine Sweeper, Repeat First, and Repeat Previous. Each value indicates the mean and standard error of ICL-Test MMER at the last step across 10 random seeds.

Method	S. Cartpole	H. L.	M. Sweeper	R. First	R. Previsous
RP(gru)	2.90 ± 0.42	0.18 ± 0.29	-1.31 ± 0.50	-4.88 ± 0.47	-7.66 ± 0.28
RP(s5)	2.87 ± 0.29	0.08 ± 0.14	-1.77 ± 0.65	-6.64 ± 0.52	-7.87 ± 0.03
FRP(gru)	$\mathbf{4.08 \pm 0.62}$	$\mathbf{2.13 \pm 1.12}$	$\mathbf{-0.69 \pm 0.56}$	$\mathbf{-4.48 \pm 0.76}$	$\mathbf{-6.04 \pm 1.65}$
FRP(s5)	3.85 ± 0.55	1.04 ± 0.82	-1.39 ± 0.38	-6.50 ± 0.40	-6.51 ± 1.10

Analysis: Does FRP's Hierarchical Bias Matter?

- Linearly Solvable MDP： with Two state space: lattice and tree
- In the **tree-structured** environment, increasing FRP depth led to a clear decrease in generalization error!(**21-15.1(tree)**> 17.5-14.8(lattice))





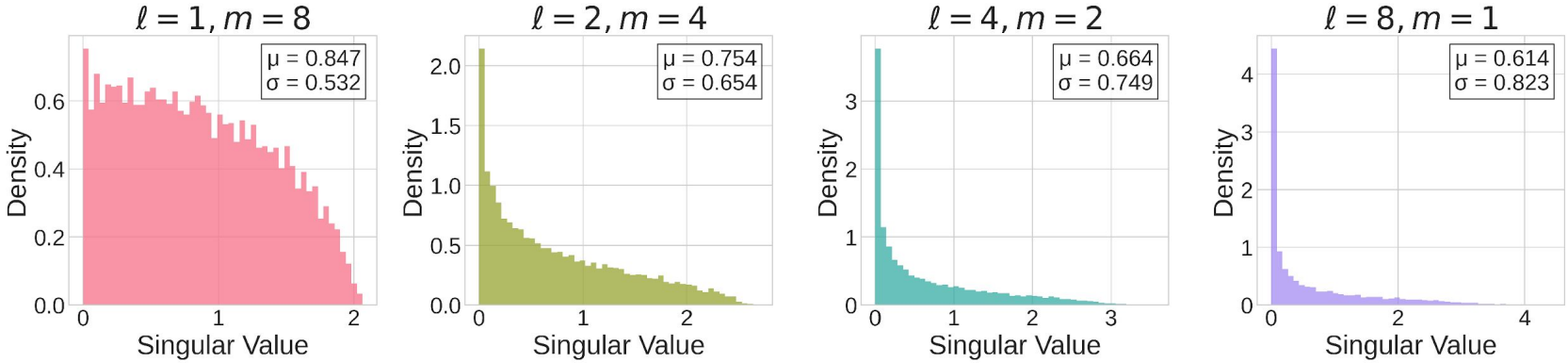
Analysis: Explanation via Effective Dimension

Effective Dimension: a measure of model complexity that appears in theoretical generalization error bounds, the smaller the effective dimension, the “simpler” the model is considered, and the better its potential to generalize. Intuitively, it represents the dimensionality of the parameter space that the model is effectively using to fit the data.

$$K^{\ell}_{i,j} = \sum_{w,w' \in \Lambda^{\ell}_n} \langle \lambda(w) X_i, \lambda(w') X_j \rangle / n_w, \qquad (4.1)$$

$(i, j = 1, 2, \ldots, p)$. The focus is on the *effective dimension* of K^{ℓ} defined as follows:

$$d_{\text{eff}}(\ell, \gamma) = \text{Tr}[K^{\ell}(K^{\ell} + \gamma I_p)^{-1}], \quad \gamma > 0. \qquad (4.2)$$



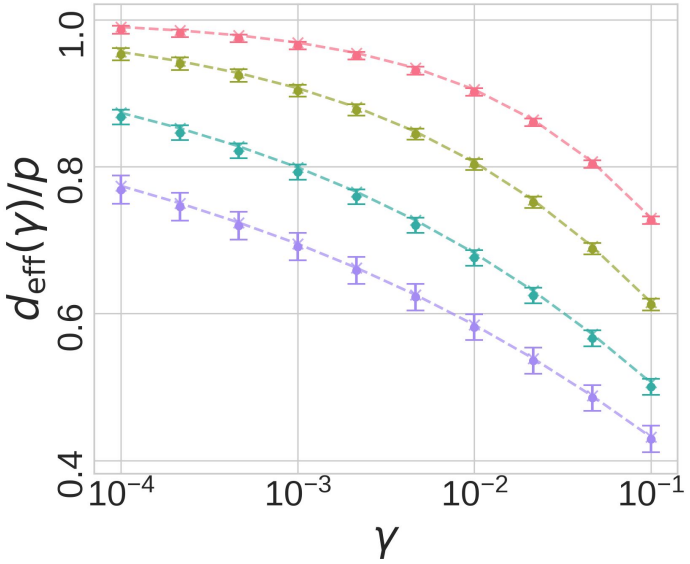
Effective Dimension

Theorem 5.1. *Fix $n, \ell \in \mathbb{N}$. Consider $\lambda : \mathbb{F}_n \to \mathrm{O}(d)$ with (3.1). Assume that $XX^\top$ has the compactly supported limit spectral distribution ν with $\int_{\mathbb{R}} t\nu(dt) \neq 0$ as $p, d \to \infty$ with $p/d \to c \in (0, \infty)$. Then, under the limit of p and d, we have*

$$\mathbb{E}[d_{\mathrm{eff}}(\gamma)/p] \to -\psi(-1/\gamma), \quad \gamma > 0, \tag{5.2}$$

where ψ is the inverse function of χ given by

$$\chi(z) = \frac{z}{z+1}\left(\frac{z/n+1}{z+1}\right)^{\ell} \mathscr{S}_\nu(z). \tag{5.3}$$



1. Introduction
2. Background : Generalization and Implicit Bias
3. Return to Math : Free Group and Random Matrices
4. Free Random Projection
5. **Future Work**

Total 41 pages