Politecnico di Torino

MASTER THESIS

Generative Modeling: Hierarchical Feature Discovery in Undersampled Structured Data

Author:
Milo REPOSSI

Supervisors:
Andrea PAGNANI
Olivier RIVOIRE

A thesis submitted in fulfillment of the requirements for the degree of Physics of Complex Systems

in the

Gulliver Lab, ESPCI

POLITECNICO DI TORINO

Abstract

Physics of Complex Systems

Generative Modeling: Hierarchical Feature Discovery in Undersampled Structured Data

by Milo REPOSSI

Generative models are widely used in protein design but remain poorly understood from a theoretical standpoint. This work aims to move beyond treating these models as black boxes by studying their internal mechanisms and learning behavior. The statistical models typically used for inference in this area are formulated as inverse problems, drawing inspiration from statistical physics to understand complex biological systems. To gain insight into how these models learn from data, we study a simplified version based on Gaussian statistics, where the ground truth is explicitly known. Using a teacher-student setup, we simulate data from a known model and evaluate how effectively another model can learn from it. This framework enables us to observe how learning progresses as more data becomes available. By applying tools from Random Matrix Theory, we uncover that features are learned in a hierarchical manner: higher-variance modes are captured earlier, while lower-variance modes require more data to be accurately inferred. This finding highlights a fundamental structure in how generative models process and prioritize information when faced with limited biological data

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Olivier Rivoire, for his unwavering availability and guidance throughout this project. His ability to support my work while giving me the freedom to explore my own ideas has been invaluable to my transition from student to researcher. I am also sincerely thankful to Marion Chauveau for his constant support, for always being there when I needed assistance, and for being a good friend during my internship. I extend my thanks to Yaakov Kleeorin, whose work served as a major inspiration for this project. Finally, I would like to thank the entire Ranganathan Lab at the University of Chicago, and in particular Madhav Mani, for the many fruitful and stimulating discussions that greatly enriched my research.

Contents

Chapter 1

Introduction

Proteins are essential biological macromolecules that perform a vast array of functions within living cells, such as providing structural support, catalyzing chemical reactions, transmitting signals, and regulating genes. These molecules can be regarded as complex *dynamical* systems as they have the ability of adapting to the environment through *evolution*. One of the aims of *quantitative biology* is to understand how these molecules fold to take their three-dimensional shape, and how they specialize to perform specific function. Approaches based on the mechanistic modeling of proteins, like molecular dynamics (MD), often fall short due to the high number of coordinates that should be tracked in order to simulate accurately these complex molecules.

In recent years, thanks to the rapid increase in sequence data availability, a *statistical* approach to the study of proteins proved very promising. In this framework one interprets a protein as a linear sequence of amino acids and tries to find the statistical patterns inside the sequence that determine its biological properties, such as structure and function. The formalism of *statistical physics* proves useful in this context, as it provides tools suited for the modeling of systems with a very large number of interacting particles, even though the space of sequences lacks the symmetries and conservation laws that typically simplify the discussion of physical systems.

More specifically, the statistical modeling of sequence data can be interpreted as an *inverse problem*, that is the problem of *inferring* which model has generated the observed data. This is the opposite of what is commonly done in physics, where one tries to derive observable properties of a system starting from the fundamental, microscopical rules that govern it. The problem of *inferring* statistical models is ubiquitous in all areas of science, and the most modern approach to this problem is that of using the techniques of *machine learning*, that is a branch of artificial intelligence that focuses on algorithms that learn from data. The field of machine learning itself has a lot in common with statistical physics, as models can often be interpreted as complex systems, allowing for the analysis of their internal mechanisms through the methods of *Disordered Systems* and *Random Matrix Theory*.

The most prominent instantiation of physics-inspired modeling of protein sequence data is DCA (Direct Coupling Analysis), which aims at fitting the available data to the equilibrium distribution of a Potts model. This choice, as we will see, is equivalent to assuming that the biologically relevant features of proteins are encoded in the single

and pairwise frequencies of the observed data. This modeling scheme is appreciated by the quantitative biology community as it provides a powerful, yet interpretable framework both for understanding how proteins fold in their three-dimensional structure (Morcos et al., 2011) and to generate novel functional sequences (Russ et al., 2020).

Nevertheless, DCA is routinely applied in a drastically *undersampled* regime, that is, the number of parameters that models are supposed to fit to data is much bigger than the number of available data points. The practical solution to this problem is that of using *regularization* techniques that make it possible to compute a unique solution to these otherwise under-constrained problems. Another problematic feature of protein sequence data is that it is *multi-scale*, that is, relevant patterns might involve just a few amino acids or they could involve large groups of cooperating units, or they could include both. The former case is typical of *contacts*, that is spots of epistatic interaction, while the latter refers to the presence of *sectors* (Halabi et al., 2009). All in all, when working with protein data we have to deal with scarce, highly structured data, and the main goal of this work is to understand **how these two properties of data interplay and affect the inference process**.

This question was recently investigated by Kleeorin et al., 2023, where the authors showed that undersampling causes the uneven representation of features at different scales, manifesting through anomalous peaks in the magnitude of the inferred parameters for critical values of the dataset size. Still, a theory explaining the phenomenology reported in this work is missing. Additionally, in the context of *supervised learning* recent works have highlighted interestingly similar phenomenology, called *double descent* (Rocks and Mehta, 2022) and *multiple descent* (Mel and Ganguli, 2021) where the test error of supervised models diverges for critical values of the dataset size due to *overfitting*.

The second goal of this work is to **bridge the theory of supervised and unsupervised models** by isolating mechanisms that are typical of learning in general and don't depend on the specific model choice.

More in general, this work fits into the bigger picture of developing a *theoretical understanding* of *how* generative models work internally, in order to go beyond black-box predictions. This is a necessary achievement as quantitative biology is becoming more and more a *data driven* discipline, and a solid theory of data and learning in a biological context is still not available.

Chapter 2

Methods and Tools

2.1 Evolution and Multiple Sequence Alignments

As noted in the Introduction, proteins change over time as a result of evolution. These changes are not random, as only those that respect the biological constraints imposed by natural selection are accepted. This simple fact can be exploited as a way of identifying relevant patterns for function and structure as those that are conserved by evolution. In practice, what is commonly done is to consider families of sequences that have evolved from a common ancestor, called *homologs*. These sequences might be very different, but still have very similar three-dimensional structure and biological functionality, which means that most of the changes induced by evolution can be regarded as noise. The task of computational biology, then, is to identify which of these changes *cannot* be considered as such, and to unveil what information is encoded in them.

Protein sequences of homologs are typically organized into *Multiple Sequence Alignments* (MSAs), which are $N \times L$ rectangular arrays

$$\mathbf{A} = (A_i^a), \quad i = 1, \ldots, L, \quad a = 1, \ldots, N$$

where each of the N rows corresponds to a sequence of length L and each column to a specific position along those sequences. Each entry A_i^a in the MSA is an element from a finite alphabet of $q = 20$ amino acids, labeled by the standard one-letter amino acid codes: `A, R, N, D, C, Q, E, G, H, I, L, K, M, F, P, S, T, W, Y, V`. An additional symbol, typically a dash `-`, is used to denote alignment gaps, bringing the effective alphabet size to $q = 21$.

Now we can define the one-point frequencies $f_i(A)$ and the two-points frequencies $f_{ij}(A, B)$ for amino acids A and B

$$f_i(A) = \frac{1}{N} \sum_{a=1}^{N} \delta_{A,A_i^a} \quad ; \quad f_{ij}(A,B) = \frac{1}{N} \sum_{a=1}^{N} \delta_{A,A_i^a} \delta_{B,A_j^a}$$

where δ is the Kronecker delta symbol.

- The one-point frequencies $f_i(A)$ quantify the *conservation* of amino acid A on the i-th site, that is, how often A appears in the i-th site in our data. As an example,

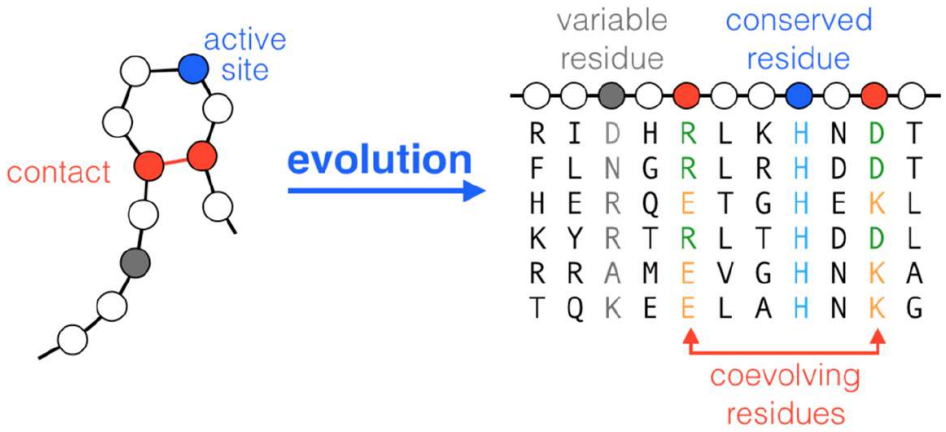

FIGURE 2.1: Co-evolution of two sites in an MSA can mean that the two sites are in contact in the three-dimensional structure of the protein. The conservation of an amino acid in some position, instead, can mean that the site is active in terms of the functionality of the molecule. This image is taken from Cocco et al., 2018.

if one amino acid is crucial for the functionality of a protein in some specific position, we expect its position to be conserved by evolution.

- The two-point frequencies $f_{ij}(A, B)$ quantify the phenomenon of *coevolution*, that is when two sites evolve together. As an example, if two sites are in contact in the folded state of the protein and the amino acid in one site changes due to a mutation, also the other amino acid will likely change to maintain the ability to form the bond. (see Figure 2.1).

It has been shown (Baldassi et al., 2014) that one and two-point frequencies are sufficient to explain a lot of amino acid variability in protein data, therefore we will use these two quantities as our main tool for extracting information from MSAs for the rest of this document.

2.2 Direct Coupling Analysis

2.2.1 Inverse problems

Our goal is to extract relevant biological information from MSAs, which falls into the very general context of *statistical inference*, that is, the art of going from raw data to usable information. The field of statistical inference is strongly related to physics, as many inference tasks can be recast as *inverse statistical physics problems*.

In physics one has a model, usually some Hamiltonian $\mathcal{H}$ describing the interactions that rule a system, and then tries to derive the macroscopic, observable properties of that system. In contrast, inverse problems work in the opposite direction: given empirical observations, the goal is to reconstruct the most likely model that could have

generated them. This is precisely what is done in the *Direct Coupling Analysis* (DCA) approach.

In DCA one assumes that the observed protein sequences are a sample of N i.i.d. [1] draws from the equilibrium distribution of a Potts model, that is a Boltzmann distribution of the type

$$P(A_1, \ldots, A_L) = \frac{1}{\mathcal{Z}} e^{-H(A_1,\ldots,A_L)}, \tag{2.1}$$

where H is the Hamiltonian of the system, and $\mathcal{Z}$ is the partition function ensuring normalization.

$$\mathcal{Z} = \sum_{A_1,\ldots,A_L} e^{-H(A_1,\ldots,A_L)}.$$

The Hamiltonian is given by

$$H(A_1, \ldots, A_L) = -\sum_i h_i(A_i) - \sum_{1 \le i < j \le L} J_{ij}(A_i, A_j), \tag{2.2}$$

where $h_i(A_i)$ are site-specific "fields" and $J_{ij}(A_i, A_j)$ are pairwise "coupling" parameters.

The Potts model

The *Potts model* is a generalization of the *Ising model* in statistical physics. The Ising model describes systems of *spins*, that is binary variables that can be in one of two states (e.g., ±1). These spins can be subject to external magnetic fields and to pairwise magnetic coupling with other spins. The Ising model provides a simple description of *ferromagnetism*. The Potts model, instead, allows each site to take on q different states, while preserving the same type of pairwise interactions plus external fields. The categorical nature of this model makes it suitable for the modeling of biological data, for example Potts variables can be amino acids when working with proteins , or *nucleotides* when modeling genomes.

Now that we have enforced the mathematical form of the distribution underlying the data, the task of learning such a distribution reduces to learning the numerical value of the fields and the couplings in Equation 2.2.

If we can develop a strategy that allows for the learning of the true distribution from which our data was generated (which we will explain in detail in subsection 2.2.3), we could then sample such a distribution in order to obtain novel sequences that were not in the original MSA. These sequences will hopefully capture the biological properties of the *training data*, potentially providing a way of generating novel sequences with a given functionality. This is the essence of *generative modeling*, which falls within the realm of *unsupervised learning*.

[1] It should be noted that biological sequence data actually show a strong bias due phylogeny, that is sequences are not actually independent draws from P, as they are all deriving from a common ancestor.

Supervised and unsupervised learning

Supervised and unsupervised learning are two fundamental approaches in machine learning. Supervised learning involves training models on labeled data, where the goal is to learn a relationship between inputs and known outputs. Unsupervised learning, on the other hand, works with unlabeled data and focuses on uncovering hidden patterns or structures. Generative modeling is a common unsupervised learning technique that aims to learn the underlying distribution of the data in order to generate new, similar samples.

2.2.2 The undersampling problem

It should be noted that for a protein sequence of length L and an amino acid alphabet of size q, in order to fully characterize the distribution in Equation 2.1 one should learn q^2L^2 couplings $J_{ij}(A, B)$ and qL external fields $h_i(A)$. For a typical sequence of length $L \approx 100$ this means that the total number of parameters of our model is $\approx 10^6$. Common sense suggests that in order to carry out the inference of these parameters in a reliable way, one should observe all the possible combinations of sites and amino acids *at least once*, that is, the number of data points should be at least as big as the size of the parameter space. Unfortunately, typical MSAs have sizes that range from a few hundreds to $N \approx 10^4$ in the luckiest scenarios. Additionally, when working with MSAs we have no guarantee that sequence space is being sampled uniformly, therefore the few samples that we have might be very concentrated in some regions, making the *undersampling* problem even more dramatic. This means that many combinations will never be observed, and therefore the *exact* inference of the distribution in Equation 2.1 is out of reach. Still, useful information is encoded in the data that we have at our disposal, and even though we cannot capture its true distribution, we can aim at estimating some *other* distribution that will hopefully capture the biologically relevant features of the true one. One of the goals of this work is to understand which of these features can be learned reliably when working with limited data, that it is in the undersampled regime.

In practice, we can temporarily sweep the undersampling problem under the carpet and infer the parameters in Equation 2.2 anyways by using a fundamental technique in machine learning called *regularization*.

2.2.3 Optimization with regularization

The problem of finding the true model underlying the data of a MSA is usually faced by maximizing the *log - likelihood* of the model with respect to the model's parameters, which in our case are the fields $h_i(A)$ and the couplings $J_{ij}(A, B)$. By doing this, the original problem of inference is recast as an optimization problem, that is finding the maximum of a scalar function. Given N observations the log-likelihood of our model

can be written as

$$\mathcal{L}(\underline{h}, \mathbf{J} \mid \mathbf{A}) = \frac{1}{N} \sum_{i \mid A^i \in \mathbf{A}} \log P(A^i \mid \underline{h}, \mathbf{J}), \tag{2.3}$$

which corresponds to

$$\mathcal{L}(\underline{h}, \mathbf{J} \mid \mathbf{A}) = -\log \mathcal{Z} + \sum_{i,A} h_i(A) f_i(A) + \sum_{i<j,\, A,B} J_{ij}(A,B) f_{ij}(A,B)$$

Now this cost function is maximized when the gradient is zero, which entails the conditions

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial h_i(A)} &= \frac{\partial \log \mathcal{Z}}{\partial h_i(A)} - f_i(A) = P_i(A \mid \underline{h}, \mathbf{J}) - f_i(A) = 0 \\ \frac{\partial \mathcal{L}}{\partial J_{ij}(a,b)} &= \frac{\partial \log \mathcal{Z}}{\partial J_{ij}(A,B)} - f_{ij}(A,B) = P_{ij}(A,B \mid \underline{h}, \mathbf{J}) - f_{ij}(A,B) = 0 \end{aligned} \tag{2.4}$$

meaning that the the log-likelihood is maximum when the single and two-point joint probabilities predicted by the model match the empirical frequencies.

It should be noted that, as pointed out in the previous section, many of the empirical frequencies that appear in the cost function Equation 2.2.3 will be 0 due to the undersampling problem, meaning that the stationary condition in Equation 2.4 translates to (assuming $f_i(A) = 0$ as an example)

$$P_i(A \mid \underline{h}, \mathbf{J}) = 0$$

which can only be achieved by sending the model's parameters to infinity due to the exponential form of the statistical distribution in Equation 2.1. This is of course a problem as we don't want the parameters of our model to diverge only because we have insufficient data. A possible solution is to modify the original cost function Equation 2.3 by adding a penalty term proportional to the magnitude of the parameters, therefore obtaining

$$\mathcal{L}(\underline{h}, \mathbf{J} \mid \mathbf{A}) = \frac{1}{N} \sum_{i \mid A^i \in \mathbf{A}} \log P(A^i \mid \underline{h}, \mathbf{J}) - \lambda R[\underline{h}, \mathbf{J}] \tag{2.5}$$

where $R[\underline{h}, \mathbf{J}]$ is some function of the model parameters and λ is called *regularization strength*. This will ensure that models with values of $\underline{h}$ and $\mathbf{J}$ that are too big will not win the optimization problem.

There are many possible choices for the function $R[\underline{h}, \mathbf{J}]$, the main two being L_1 and L_2 regularization:

- with L_1 regularization one chooses a penalty that is proportional to the absolute value of the parameters

$$R_{L_1}[\underline{h}, \mathbf{J}] = \sum_{i<j,a,b} \left| J_{ij}(a,b) \right| + \sum_{i,a} |h_i(a)|$$

- with L_2 regularization, instead, the penalty is proportional to the square of the parameters

$$R_{L_2}[\underline{h}, \mathbf{J}] = \sum_{i<j,a,b} \left|J_{ij}(a,b)\right|^2 + \sum_{i,a} |h_i(a)|^2$$

In order to solve this optimization problem we need to find the set of parameters that will maximise the Likelihood in Equation 2.5. This operation typically cannot be done analytically and requires a numerical optimization technique called *gradient descent*.

Gradient Descent

Gradient descent (GD) is an iterative optimization algorithm used to minimize a differentiable function. Given a function $f(\theta)$, the algorithm updates the parameters θ in the direction of the negative gradient of the function at the current point. The update rule is given by:

$$\theta^{(t+1)} = \theta^{(t)} - \alpha \nabla f(\theta^{(t)}),$$

where $\alpha > 0$ is the learning rate and $\nabla f(\theta^{(t)})$ is the gradient of f evaluated at $\theta^{(t)}$. This process is repeated until convergence. It should be noted that there exist many variants of this algorithm. The version that we report here (which is also the one that we will use in Appendix A) is its simplest version, often referred as *vanilla* gradient descent.

Once gradient descent has converged, we are left with a set of parameters

$$[\underline{h}^*, \mathbf{J}^*] = \arg\max_{h,J} \{\mathcal{L}(\underline{h}, \mathbf{J} \mid \mathbf{A})\} \tag{2.6}$$

which fully characterize the maximum-likelihood model that generated our data.

2.2.4 The teacher-student setting

As explained in section 2.2 the goal of protein sequence modeling is to learn the true model that generated the observed data. When working with natural data, though, we have to deal with two layers of uncertainty:

1. A *ground truth* model is not available. We are *enforcing* the functional form of the distribution we wish to learn, but it might be the case that the distribution is not expressive enough to capture the actual structure of the data.

2. The undersampling problem described in subsection 2.2.2 prevents the exact learning of such a distribution, which might not be correct in the first place.

We want to understand the influence of undersampling on learning, therefore it is desirable to isolate its effects by working in a scenario in which at least we know the ground truth distribution, thus mitigating the problematics related to (1). The most

common way to do this is to use the *teacher-student* framework. In this framework one has two models, called respectively *teacher* and *student*, then produces data by sampling the teacher model, and then has the student model trying to infer the parameters of the teacher model from the generated data. This framework works well for our goal as it will allow us to decide *how much* data is given to the student model (which means we can "tune" the amount of undersampling) while knowing the ground truth distribution of the generated data, allowing us to

1. Quantify *how wrong* the student model is with respect to the teacher.
2. Be sure that the student model is able to learn the teacher model, when given enough data, as the two models have the same functional form.

2.2.5 Limited data with structure - a small literature review

In subsection 2.2.2 we have discussed how DCA is normally deployed in a strongly undersampled regime, that is the number of data points is usually much smaller than the number of parameters to be inferred. Still, DCA proves powerful in predicting some features of natural sequence data despite this limitation, for example by predicting accurately the presence of contacts (see Morcos et al., 2011). This suggests that the influence of undersampling is not uniformly distributed across features, but seems to impact some more than others for a given MSA size N. A systematic exploration of the uneven representation of features was carried out by Kleeorin et al., 2023 by using DCA in a teacher-student setting. The authors chose a teacher model that presented different types of structures in its true couplings matrix $\mathbf{J}$. More in detail, as it can be seen in Figure 2.2:

- 3 *contacts*, that is isolated spots of pairwise interaction. In biological terms these structures can be thought of as sites of epistatic interaction,
- a small *sector*, that is a small collective of coupled neighboring sites,
- a bigger *sector*.

The authors show that the value of the inferred parameters $\hat{\mathbf{J}}$ through DCA with regularization exhibit anomalous peaks as a function of the MSA size N. The positions of these peaks depend on the considered structure, as well as their sharpness (see Figure 2.2). More in particular, these peaks show up in a hierarchical way: the first one to appear is associated to contacts, followed by the small sector, in turn followed by the big sector.

This work constitutes important evidence of the fact that structure in the input data impacts in some non-trivial way *how well models learn* different features for a given size of the training dataset.

The phenomenology described above in context of DCA models shows some similarities with a well known phenomena in supervised learning called *double descent*. This phenomenon consists of the emergence of a peak in the test error of supervised

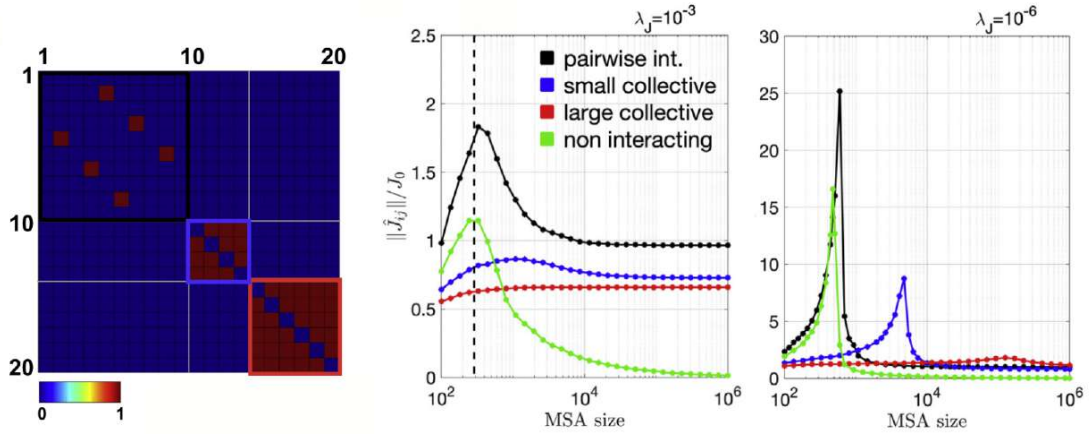

FIGURE 2.2: Anomalous peaks in the inference of three different structures in the input data. On the left: the matrix of couplings of teacher model. On the right: the value Frobenius norm of the inferred couplings $|| J_{ij} ||$ as a function of the MSA size N, for two different values of regularization strength $\lambda = 10^{-3}$, $\lambda = 10^{-6}$. Lower regularization makes the peaks sharper and the separation of their positions more dramatic. These pictures are taken from Kleeorin et al., 2023.

models in correspondence of the critical value of $\alpha = \frac{N}{P} = 1$ where N is the number of data points the model is fed, and P is the complexity of the model.

This phenomena was studied by Rocks and Mehta, 2022 through the lens of statistical physics by using the *Cavity Method* of disordered systems, and is well understood as an instance of *overfitting*, that is when models interpolate noise in the training data therefore losing the ability to generalize well. The authors use a simple supervised task called *Ridge regression* as a playground, and assume that the input data the model is fed is distributed according to an isotropic Gaussian distribution.

Ridge Regression

Linear regression is a method used to model the relationship between a dependent variable and a set of independent variables by fitting a linear equation to observed data. The model assumes a linear form:

$$\hat{y} = w_0 + \sum_{j=1}^{p} x_j \hat{w}_j$$

where $\hat{y}$ is the predicted value, x_j are the input features, and $\hat{w}_j$ are the learned coefficients. *Ridge regression* extends linear regression by adding L_2 regularization. The optimal weight vector $\hat{\mathbf{w}}$ that minimizes the prediction error is given by the normal equation:

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X} + \lambda \mathbb{I})^{-1} \mathbf{X}^\top \mathbf{y}$$

where λ is the regularization strength, $\mathbf{X}$ is our data array, and $\mathbf{y}$ is the vector of labels.

Recently, Mel and Ganguli, 2021 investigated how structure in the input data impacts the double descent peak, and the phenomena of *multiple descent* was unveiled. More in detail, the authors chose to work with Gaussian input data that had two widely separated variance scales, that is a *high-variance* mode and a *low-variance* mode, as showed in Figure 2.3. The result is that the generalization error exhibits peaks not just at the *interpolation threshold* $\alpha = 1$, but also for a lower value of α which depends on how the modes are split in the covariance matrix $\mathbf{\Sigma}$ of the input data. As an example, considering a P-dimensional regression task, if we choose $\frac{P}{2}$ high variance modes and $\frac{P}{2}$ low variance modes we obtain the usual peak at $\alpha = 1$, with the addition of a new peak at $\alpha = 0.5$ (see Figure 2.3).

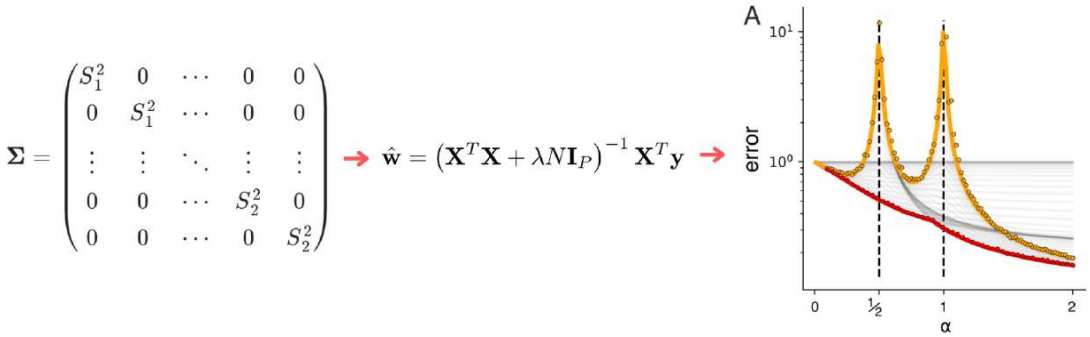

FIGURE 2.3: Multiple Descent in Ridge Regression. This image is taken from Mel and Ganguli, 2021.

It should be noted that the peaks observed in Kleeorin et al., 2023 and those observed in Mel and Ganguli, 2021 have a fundamentally different nature. In particular, the former involve the value of the *inferred parameters*, while the latter involve the test

error of the model, a quantity that has no counterpart in the unsupervised setting. Still, the two phenomenologies present some common traits:

- The position of the peaks depends on the structure of the input data.
- The sharpness of the peaks depends on the structure and on regularization strength λ.

In this work we aim at finding a common ground between these two settings, providing an explanation of the peaks observed in Kleeorin et al., 2023 by working in the limit of widely separated scales as done in Mel and Ganguli, 2021. As we explain in the next section, such a common ground is represented by *Gaussian DCA* in a teacher-student setting.

2.3 Gaussian DCA

The main goal of this work is to provide an understanding of how models learn different features in the undersampled regime, working along the lines of Kleeorin et al., 2023. In order to this, it is desirable to work in a simplified scenario with respect to the classical DCA framework described above for two main reasons:

- It will allow us to pinpoint the true origin of the phenomena observed in Kleeorin et al., 2023, ruling out the idea that the observed phenomenology is exclusive of the specific chosen model or the chosen regularization scheme.
- If we simplify the model enough, it will be possible to perform analytical calculations.

2.3.1 Learning a Gaussian model with Random Matrix Theory

The first important simplification that we will do is to switch from "classical" DCA to *Gaussian DCA*. This means that instead of modeling discrete categorical variables (e.g., amino acids) using a Potts Hamiltonian, we will assume that each sequence is a real-valued vector $\mathbf{x} \in \mathbb{R}^P$ drawn from a multivariate Gaussian distribution

$$p(\mathbf{x}) = \frac{|\mathbf{J}|^{1/2}}{(2\pi)^{P/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \mathbf{J}(\mathbf{x}-\boldsymbol{\mu})\right) \tag{2.7}$$

where $\mathbf{J} = \boldsymbol{\Sigma}^{-1}$ is the $P \times P$ precision matrix containing the pairwise couplings and $\boldsymbol{\Sigma}$ is the corresponding *covariance matrix*. For simplicity, in the following we will assume that $\boldsymbol{\mu} = 0$. It should be noted that Gaussian models can be interpreted as the mean-field approximation of Potts models, as we discuss in Appendix B.

Now we can deploy Gaussian models in the teacher-student setting introduced in subsection 2.3.1 by defining the "teacher model" with known parameters $\mathbf{J_t}$

$$p_t(\mathbf{x}) = \frac{|\mathbf{J_t}|^{1/2}}{(2\pi)^{P/2}} \exp\left(-\frac{1}{2}\mathbf{x}^\top \mathbf{J_t}\mathbf{x}\right) \tag{2.8}$$

and we can form a dataset by drawing N i.i.d. samples from $p_t(\mathbf{x})$ which we stack in a $N \times P$ matrix $\mathbf{X}$. Let us now define the normalized $P \times P$ *empirical covariance matrix* $\mathbf{S} = \frac{1}{N}\mathbf{X}^\mathbf{T}\mathbf{X}$ and the *sampling ratio* $\alpha = \frac{N}{P}$. We can now compute from Equation 2.3 the log-likelihood of the coupling matrix $\mathbf{J}$ given the data $\mathbf{X}$, adding a $\tilde{L}_1$ *spectral penalty* term $R[\mathbf{J}] = \mathrm{Tr}[\mathbf{J}]$ as done in Catania et al., 2025:

$$\log \mathcal{L}(\mathbf{J} \mid \mathbf{X}) = \frac{1}{2}\log\det(\mathbf{J}) - \frac{1}{2}\mathrm{Tr}[\mathbf{J}\mathbf{S}] - \lambda\,\mathrm{Tr}[\mathbf{J}] \tag{2.9}$$

This regularization choice is uncommon but convenient, as it leads to a closed-form expression for the maximum-likelihood estimator of the coupling matrix $\hat{\mathbf{J}}$:

$$\hat{\mathbf{J}} = \left[\frac{1}{N}\left(\mathbf{X}^T\mathbf{X}\right) + 2\lambda\mathbb{I}\right]^{-1} \tag{2.10}$$

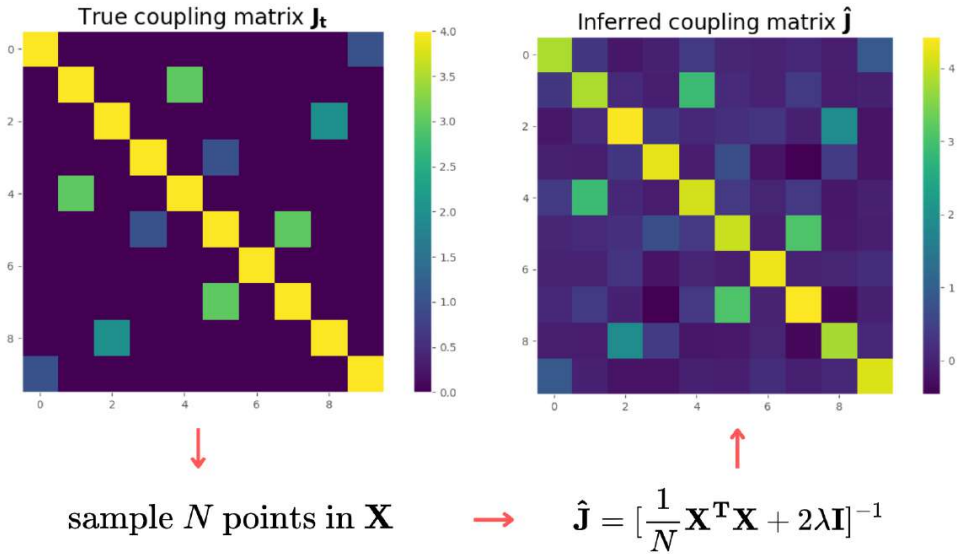

FIGURE 2.4: Teacher-student setting using $\tilde{L}_1$ regularization in a Gaussian setting. $P = 10$ and $\lambda = 10^{-3}$. Here we chose $\alpha = \frac{N}{P} = 10$, placing us in the well-sampled regime. Not surprisingly, the inferred coupling matrix closely resembles the true one.

which coincides with $G_\mathbf{S}(-2\lambda)$ where $G_\mathbf{S}$ is the resolvent of the empirical covariance matrix. We can give a Bayesian interpretation to this estimator by observing that it assigns a *prior* to inferred couplings $\hat{\mathbf{J}} = \frac{1}{2\lambda}\mathbb{I}$ when data is absent or scarce, then this prior is updated as more data is added and the rank of $\mathbf{S}$ increases. It should be noted that in the undersampled regime $\alpha < 1$ the empirical covariance matrix $\mathbf{S}$ is rank-deficient and therefore not invertible, but the addition of $+2\lambda\mathbb{I}$ makes its columns linearly independent ensuring invertibility.

Resolvent of a matrix

Given an $N \times N$ real symmetric matrix $\mathbf{A}$, its resolvent is given by $\mathbf{G}_{\mathbf{A}}(z) = (z\mathbf{1} - \mathbf{A})^{-1}$

All in all, we have reduced the original inference problem of minimizing the cost function Equation 2.9 to that of computing the empirical covariance matrix $\mathbf{X}^\top \mathbf{X}$ and applying the simple transformation prescribed in Equation 2.10 in order to obtain the optimal coupling matrix $\hat{\mathbf{J}}$.

When the true model underlying the data contained in $\mathbf{X}$ is simple enough (see subsection 3.0.1 and subsection 3.0.2), it is often possible to characterize the spectral properties of $\mathbf{X}^\top \mathbf{X}$, which in turn can be propagated to those of the inferred coupling matrix $\hat{\mathbf{J}}$. This can be done by using the tools of *Random Matrix Theory* (RMT) in the "thermodynamic limit" $N, P \to \infty$, $\alpha = \frac{N}{P}$. We will show with simulations in subsection 3.0.1 that α is indeed the relevant scaling variable for this problem.

Random Matrix Theory

Random Matrix Theory (RMT) is a field of mathematics and physics that studies the properties of matrices whose entries are random variables. It originated in the 1950s from the work of physicist Eugene Wigner, who used it to model the energy levels of heavy atomic nuclei, where exact solutions were intractable. The central focus of RMT is on the spectral properties of large random matrices—specifically, the distribution and behavior of their eigenvalues and eigenvectors as the matrix size tends to infinity.

Chapter 3

Results

In this section we will characterize the maximum likelihood coupling matrix $\hat{\mathbf{J}}$ derived in Equation 2.10 by working with teacher models that present no interactions, that is $\mathbf{J_t}$ is a diagonal matrix. We will deal both with the case of an isotropic coupling matrix and that of a matrix with many widely separated modes.

3.0.1 Inferring an isotropic matrix

Let us choose a ground truth coupling matrix of the type $\mathbf{J_t} = J_0\mathbb{I}$ with $J_0 \in \mathbb{R}$. This scenario is particularly relevant since the empirical covariance matrix $\mathbf{S}$ is a (scaled) Wishart matrix in the $N, P \to \infty$ limit (Potters and Bouchaud, 2021).

Spectral Density

For a given random matrix $\mathbf{A}$, we can define the empirical spectral distribution (ESD) also called the sample eigenvalue density:

$$\rho_N(\lambda) = \frac{1}{N}\sum_{k=1}^{N}\delta\left(\lambda - \lambda_k\right)$$

where λ_k are the eigenvalues of $\mathbf{A}$. When $N \to \infty$ the ESD converges to the *spectral distribution* $\rho(\lambda)$.

Wishart matrices play a central role in RMT and their *spectral distribution* is given by the *Marchenko-Pastur* (MP) law (Potters and Bouchaud, 2021):

$$\rho(x) = \begin{cases} (1-\alpha)\,\delta(x) & \text{if } \alpha < 1 \text{ and } x = 0, \\ \dfrac{J_0\sqrt{(x_+ - x)(x - x_-)}}{2\pi\alpha x} & \text{if } x \in [x_-, x_+], \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

where $x_\pm = \frac{1}{J_0}\left(1 \pm \sqrt{\frac{1}{\alpha}}\right)^2$ and $\alpha = \frac{N}{P}$. As one can read from Equation 3.1 this distribution includes both a continuous part with support on $[x_-, x_+]$ and a δ mass at $x = 0$, which is only present when $\alpha < 1$.

We want to investigate how well the estimator in Equation 2.10 (that is, our *student model*) reproduces $\mathbf{J_t} = J_0\mathbb{I}$ as the number of data points N sampled from the *teacher model* increases at fixed P.

As it is done in RMT, we will consider $\hat{\mathbf{J}}$ as an array of random variables, and we will compute the *mean* and the *variance* of its entries. We present in Appendix B an argument to support the fact that the phenomenology observed in Kleeorin et al., 2023 should be found in this context by looking at the variance of the elements of $\hat{\mathbf{J}}$ instead of their mean value.

First of all we can inspect the eigenvalues of $\hat{\mathbf{J}}$, which are given by

$$\hat{J}_\alpha = \frac{1}{s_\alpha + 2\lambda} \tag{3.2}$$

where s_α is the α-th eigenvalue of $\mathbf{S} = \frac{1}{N}\mathbf{X}^T\mathbf{X}$ and $\hat{J}_\alpha$ is the α-th eigenvalue of $\hat{\mathbf{J}}$. We can easily find that the average estimated eigenvalue $\langle \hat{J} \rangle = \frac{1}{P}\sum_\alpha \hat{J}_\alpha$ is given by the Stiltjes transform of the Wishart ensamble

$$\mathbb{E}[\hat{J}] = g(-2\lambda) \tag{3.3}$$

where $\mathbb{E}[\cdot]$ denotes the expected value and the expression of the Stiltjes transform is given by

Stiltjes Transform

The *Stiltjes transform* of a matrix $\mathbf{A}$ is given by

$$g_N^{\mathbf{A}}(z) = \frac{1}{N}\operatorname{Tr}\left(\mathbf{G}_{\mathbf{A}}(z)\right) = \frac{1}{N}\sum_{k=1}^{N}\frac{1}{z-\lambda_k}$$

where $\mathbf{G}_{\mathbf{A}}$ is the resolvent of $\mathbf{A}$ and λ_k are the eigenvalues of $\mathbf{A}$. If we call ρ_N the empirical spectral distribution of $\mathbf{A}$ then the Stiltjes transform can be written as

$$g_N(z) = \int_{-\infty}^{+\infty}\frac{\rho_N(\lambda)}{z-\lambda}\mathrm{d}\lambda \xrightarrow{N\to\infty} g(z) = \int_{-\infty}^{+\infty}\frac{\rho(\lambda)}{z-\lambda}\mathrm{d}\lambda$$

$$g(z) = \frac{\sigma^2(1-q) - z - \sqrt{\left(z-\sigma^2(q+1)\right)^2 - 4q\sigma^4}}{2qz\sigma^2} \tag{3.4}$$

where $\sigma^2 = J_0^{-1}$ and $q = \alpha^{-1}$.

We can see in Figure 3.1 (A) that the formula above fits well the simulation data even for finite P. The model drastically overestimates the coupling J_0 in the under-sampled regime, while the inferred value becomes accurate after the critical value of $\alpha = 1$. The interpretation of this result is straightforward: when $\alpha \to 0$ we find that $\mathbf{S}$

is low rank, meaning that most of his eigenvalues s_α are equal to 0, and the eigenvalues of $\hat{\mathbf{J}}$ are those of the prior, that is $(2\lambda)^{-1}$.

When α increases, instead, more and more of the eigenvalues of $\mathbf{S}$ accumulate in the continuous part of the MP distribution (see Equation 3.1), causing the average eigenvalue to decrease like $\frac{1-\alpha}{\lambda}$ until when, at $\alpha = 1$, the δ mass in 0 of the MP distribution is finally depleted.

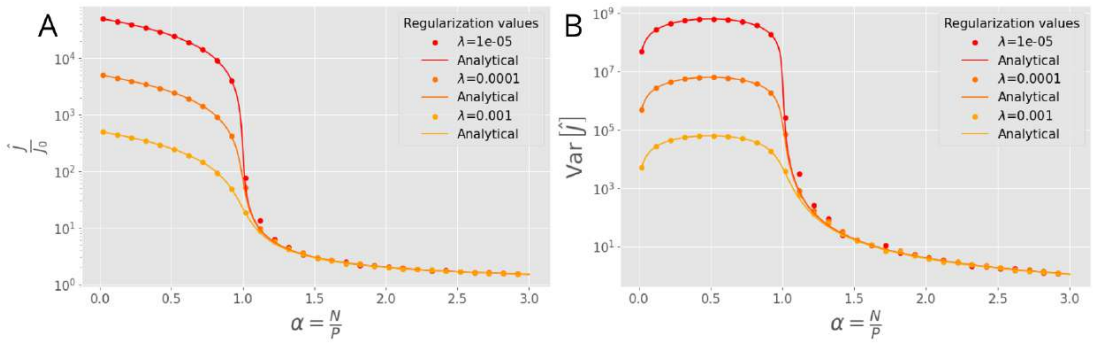

FIGURE 3.1: (A) Mean and (B) variance of the eigenvalues of $\hat{\mathbf{J}}$ as a function of $\alpha = \frac{N}{P}$ for different regularization values. $P = 100$ and $J_0 = 1$. Solid lines are the theoretical curves while scatter points are the simulation data.

Now we turn our attention to the variance of the inferred eigenvalues. We can compute it directly from Equation 3.2 obtaining

$$\text{Var}[\hat{J}] = g'(-2\lambda) - g(-2\lambda)^2 \tag{3.5}$$

where $g(z)$ was defined in Equation 3.4 and $g'(z)$ is its first derivative with respect to z.

We can see in Figure 3.1 (B) that the variance of the inferred couplings is very low in the $\alpha \to 0$ limit, peaks for a critical value of $\alpha^* = 0.5$, and then decreases abruptly again at $\alpha = 1$. **This phenomena is the simplest instance of the mechanism underlying the peaks observed in Kleeorin et al., 2023.**

We can obtain a simplified description of the curves in Figure 3.1 (B) by expanding Equation 3.5 in the low regularization limit $\lambda \to 0$ when $\alpha < 1$ by only considering the contributions to $g(z)$ given the δ mass at zero from the MP distribution. We have

$$g(z) \approx \frac{1-\alpha}{z} \quad ; \quad g'(z) \approx \frac{1-\alpha}{z^2}$$

which yields

$$\text{Var}[\hat{J}] \approx \frac{-\alpha^2 + \alpha}{4\lambda^2} \tag{3.6}$$

meaning the shape of the variance curve is approximately parabolic in the undersampled ($\alpha < 1$) regime.

It is natural to ask what is special about the value of $\alpha^* = 0.5$ for which the parabola in Equation 3.6 has its maximum. We can develop an intuition for this by inspecting the histogram of the eigenvalues of $\hat{\mathbf{J}}$ in Figure 3.2. We can see that when $\alpha < 1$ their distribution is bimodal, exhibiting a δ-peak at $\hat{J} = (2\lambda)^{-1}$ and a bulk centered in $\hat{J} = J_0$. When $\alpha \to 0$ all the eigenvalues belong to the peak on the right, while at $\alpha \to 1$ this peak depletes and all the mass in accumulated in the bulk around J_0.

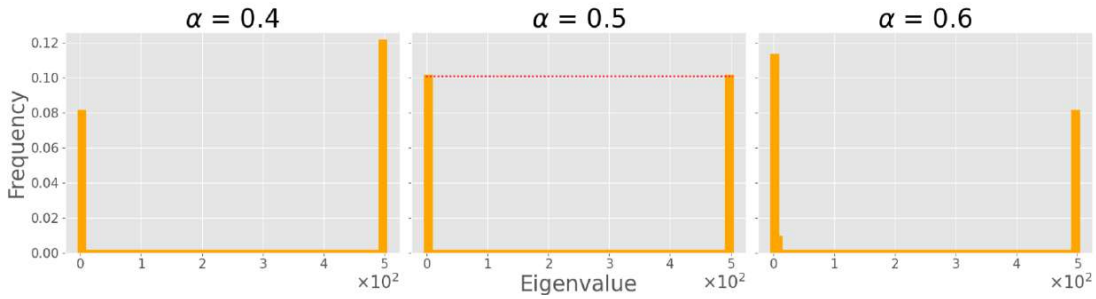

FIGURE 3.2: Histogram of the eigenvalues of $\hat{\mathbf{J}}$ close to the critical value $\alpha^* = 0.5$. $P = 100$, $\lambda = 10^{-3}$, $J_0 = 1$. It should noted that the right bin centered on $(2\lambda)^{-1}$ is a δ-peak, while the left bin centered on J_0 has a non-trivial bulk shape.

The value of $\alpha^* = 0.5$ is therefore the special value for which the two bins have the same height. Intuitively, we can think of this point as the value of α for which the model is *maximally uncertain* about its prediction. It should be noted that the expansion in Equation 3.6 holds only if J_0 is close to zero so that the distance between the two bulks is $\approx (2\lambda)^{-1}$.

The position of the peak can be changed by tuning the two parameters of this model (J_0, λ). In particular, as its shown in Figure 3.3 (B), the peak can only be pushed to the left (that is towards $\alpha = 0$) by increasing J_0 and λ.

It should be noted that the statistical distribution of the eigenvalues of $\hat{\mathbf{J}}$ depends only on the ratio $\alpha = \frac{N}{P}$, proving that the limit $N, P \to \infty$ and $\alpha \approx 1$ is the right scaling limit for this problem as we argued in subsection 2.3.1. We can check that this is actually the case by computing numerically the eigenvalues of $\hat{\mathbf{J}}$ for different values of P and observing that they lie on the same curve (see Figure 3.3 (A)).

Now we can investigate how the statistical properties of the eigenvalues of $\hat{\mathbf{J}}$ propagate to its matrix elements $\hat{J}_{ij}$. We start by noting that $\mathbb{E}[\hat{J}_{ij}] = \mathbb{E}[\mathbf{G}_{ij}] = 0$ for $i \neq j$ due to rotational invariance of the resolvent. Still, we can compute the variance of the off-diagonal entries of $\hat{\mathbf{J}}$ exploiting the spectral decomposition

$$\hat{J}_{ij} = \sum_{k=1}^{P} U_{ik} J_k U_{jk} \tag{3.7}$$

where J_k is the k-th eigenvalue of $\hat{\mathbf{J}}$ and U is the matrix of the eigenvectors of $\mathbf{S}$, which follows the Haar distribution.

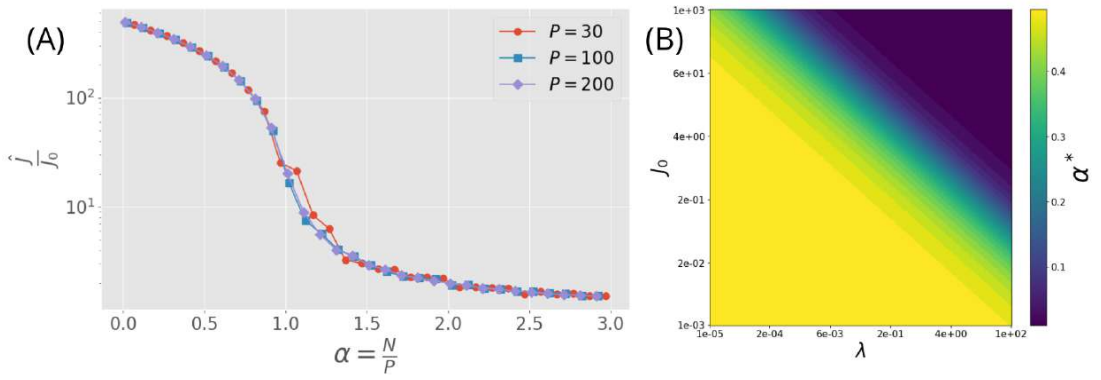

FIGURE 3.3: On the left (A): learning of the eigenvalue of $\mathbf{J_t}$ as a function of α for problems of different size P. On the right (B): heatmap of the position of the peak α^* as a function of the coupling strength J_0 and regularization strength λ.

The Haar measure

If W is a real Wishart matrix of size $P \times P$ (with parameters already specified), write its spectral decomposition as

$$W = U \Lambda U^T,$$

where $U \in O(P)$ is an orthonormal matrix of eigenvectors and $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_P)$ are the eigenvalues.
Then U is distributed according to the (normalized) Haar measure on $O(P)$, i.e. for any fixed $V \in O(P)$,

$$P(U) = P(VU) = P(UV),$$

and furthermore U is independent of Λ.

By using the 4-point correlation function of Haar matrices (Potters and Bouchaud, 2021 - Sec. 12.1.2) together with the fact that the eigenvalues and the eigenvectors of a Wishart matrix are independent, we obtain

$$\text{Var}[\hat{J}_{ij}] = \frac{1}{P}(\mathbb{E}[\hat{J}^2] - \mathbb{E}[\hat{J}]^2) = \frac{1}{P}\text{Var}[\hat{J}] \quad (i \neq j) \tag{3.8}$$

that is, the variance of the off-diagonal matrix elements exhibits the same parabolic behavior of the eigenvalues found in Equation 3.6 but its amplitude is dampened by a factor $\frac{1}{P}$.

As for the diagonal elements of $\hat{\mathbf{J}}$, we have that $\mathbb{E}[\hat{J}_{ii}] = \mathbb{E}[\hat{J}]$, while for the variance we can still resort to the spectral decomposition Equation 3.7. By recalling that

$\mathbb{E}\left[U_{ik}^4\right] = \frac{3}{P(P+2)}$ and $\mathbb{E}\left[U_{ik}^2 U_{im}^2\right] = \frac{1}{P(P+2)}$ when $k \neq m$ we obtain

$$\mathrm{Var}[\hat{J}_{ii}] = \frac{2}{P}(\mathbb{E}[\hat{J}^2] - \mathbb{E}[\hat{J}]^2) = \frac{2}{P}\,\mathrm{Var}[\hat{J}] \tag{3.9}$$

It should be noted that we have described the distribution of the eigenvalues of $\hat{\mathbf{J}}$ in terms of their mean and variance, which is generally not an informative representation of a bimodal distribution like the one showed in Figure 3.2. Still, we have used these two quantities to compute mean and variance of the matrix entries $\hat{J}_{ij}$, which follow a Gaussian distribution (see Figure 3.4) which is indeed fully characterized by its first two moments.

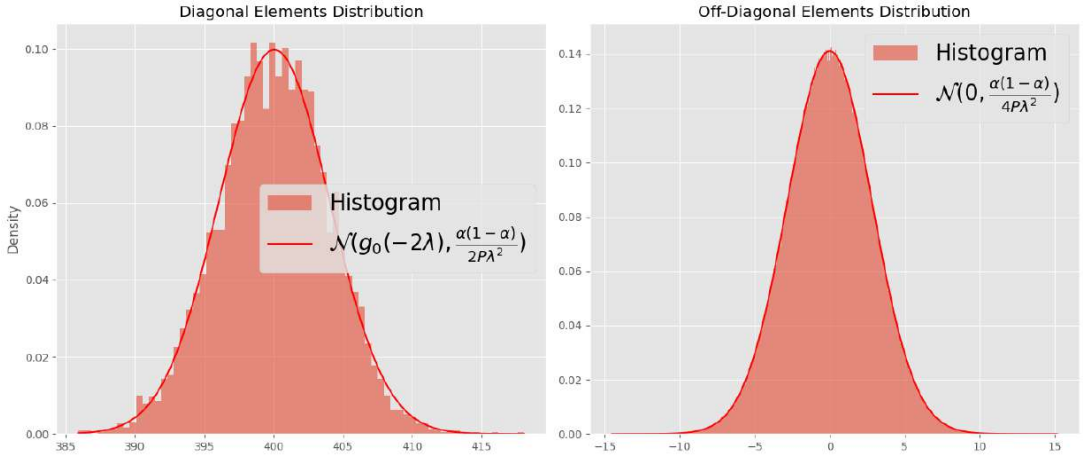

FIGURE 3.4: Statistical distribution of the diagonal and off-diagonal entries of $\hat{\mathbf{J}}$. The first distribution is centered in $\hat{J} = 0$ as expected, while the second one is centered in $\hat{J} = g(-2\lambda)$. Simulation parameters are $J_0 = 1$, $P = 5000$, $\lambda = 10^{-3}$, $\alpha = 0.2$.

As a conclusion to this section, we can say that we have provided an explanation for the simplest instance of the peaks observed in Kleeorin et al., 2023 in the context of a non-interacting isotropic model. The basic mechanism underlying this phenomena is the interplay between data and regularization, causing a peak in the variance of the inferred parameters for an intermediate value of $\alpha \in (0, 0.5]$.

3.0.2 Inferring a two modes matrix

In this chapter we will deploy the same framework of the previous section to infer a coupling matrix that includes two widely separated coupling modes. We will work along the lines of Mel and Ganguli, 2021 extending their results in the context of ridge regression to the case of Gaussian DCA, allowing us to understand how structure in the input data modifies the variance peak we have just highlighted in the isotropic

case. The true model is given by:

$$\mathbf{J_t} = \begin{pmatrix} J_1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ 0 & \cdots & J_1 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & J_2 & \cdots & 0 \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & J_2 \end{pmatrix}$$

We can tune the values of J_1,J_2 and their relative populations along the diagonal by setting P_1 entries equal to J_1 and P_2 entries equal to J_2, keeping $P = P_1 + P_2$. It should be noted that this precision matrix is associated to a covariance matrix $\mathbf{\Sigma_t} = \mathbf{J_t}^{-1}$ having variances $S_1 = J_1^{-\frac{1}{2}}$ and $S_2 = J_2^{-\frac{1}{2}}$. From now on we will make the assumption that $J_1 \ll J_2 \Rightarrow S_1 \gg S_2$, meaning that the first mode is associated to *high variance* while the second mode is associated to *low variance*.

As prescribed by the teacher-student machinery defined in subsection 2.3.1 we will draw N samples from $p_t(\mathbf{x})$, form the empirical covariance matrix $\mathbf{S} = \frac{1}{N}\mathbf{X^T X}$ and compute $\mathbf{J_t}$ through Equation 2.10. As in the isotropic case, the easiest place to start is the eigenvalues of $\hat{\mathbf{J}}$.

In the limit of widely separated scales, the spectral distribution of $\mathbf{S}$ is given by the superposition of two Marchenko Pastur distributions ρ_1 and ρ_2 roughly centered on S_1^2 and S_2^2 (Mel and Ganguli, 2021):

$$\begin{aligned} \rho_d(x) &= \frac{\sqrt{(x_+ - x)(x - x_-)}}{2\pi S_d^2 x} \\ x_\pm &= S_d^2 \left(1 - \frac{1}{\alpha}\sum_{d'<d} f_{d'}\right)\left(1 \pm \sqrt{\frac{f_d}{\alpha - \sum_{d'<d} f_{d'}}}\right)^2 \end{aligned} \tag{3.10}$$

Where $f_d = \frac{P_d}{P}$, $S_d = J_d^{-\frac{1}{2}}$ and $d = 1, 2$.

It should be noted that the spectral density of the empirical covariance matrix $\mathbf{S}$ associated to the second mode ρ_2 exists only for $\alpha > f_1 = \frac{P_1}{P}$, meaning that the spectrum of $\mathbf{S}$ goes through a phase transition at which a new "bulge" is enucleated at the critical value of $\alpha^* = f_1$, as it can be seen in Figure 3.5.

As in the previous case, we are interested in the *mean* and *variance* of the inferred eigenvalues, and thanks to the wide separation assumption we can treat the two modes separately. For the high-variance mode associated to J_1:

$$\mathbb{E}[\hat{J}_1] = \int dx \frac{\rho_1(x)}{x + 2\lambda} = g_1(-2\lambda)$$

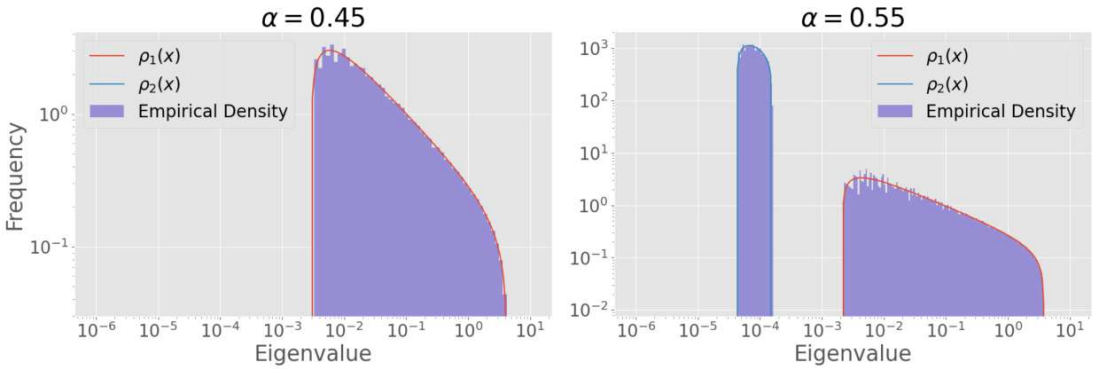

FIGURE 3.5: Distribution of the eigenvalues of $\mathbf{S} = \frac{1}{N}\mathbf{X}^\top\mathbf{X}$ when data is drawn from a model with two widely separated scales. For $\alpha < 0.5$ only the bulk relative to the first mode is visible, but when $\alpha > 0.5$ a new the one relative to the second mode also appears. The two bulges are roughly centered on S_1^2 and S_2^2 as expected. Simulation parameters are $P = 5000$, $f_1 = f_2 = 0.5$, $S_1 = 1$ and $S_2 = 10^{-2}$.

where g_1 is a "modified Stiltjes transform", directly obtained from Equation 3.4 by mapping

$$\sigma_1^2 = S_1^2, \quad q_1 = \frac{f_1}{\alpha} \tag{3.11}$$

For the low variance modes instead, we have to keep in mind that the spectral distribution has no mass if $\alpha < f_1$, in which case all the eigenvalues relative to this mode are going to be $(2\lambda)^{-1}$. When $\alpha > f_1$ instead, we can write

$$\mathbb{E}[\hat{f}_2] = \int dx \frac{\rho_2(x)}{x + 2\lambda} = g_2(-2\lambda)$$

where g_2 is obtained by mapping

$$\sigma_2^2 = S_2^2\left(1 - \frac{1}{\alpha}f_1\right), \quad q_2 = \frac{f_2}{\alpha - f_1} \tag{3.12}$$

The picture that we obtain (as it can be seen in Figure 3.6) is that the low variance modes are "frozen" to the value $\hat{f}_2 = (2\lambda)^{-1}$ set by the prior for $\alpha < f_1$, while they are get *unlocked* as soon as $\alpha > f_1$ and then start approaching their true value. The high variance modes instead are always "active" and start being learned from $\alpha = 0$ on. **The model is learning features hierarchically as a function of N, prioritizing high variance directions over low variance ones**.

Now, following the discussion of the previous section we wish to investigate the variance of the estimated eigenvalues. We can simply extend the result found in 3.0.1 by obtaining

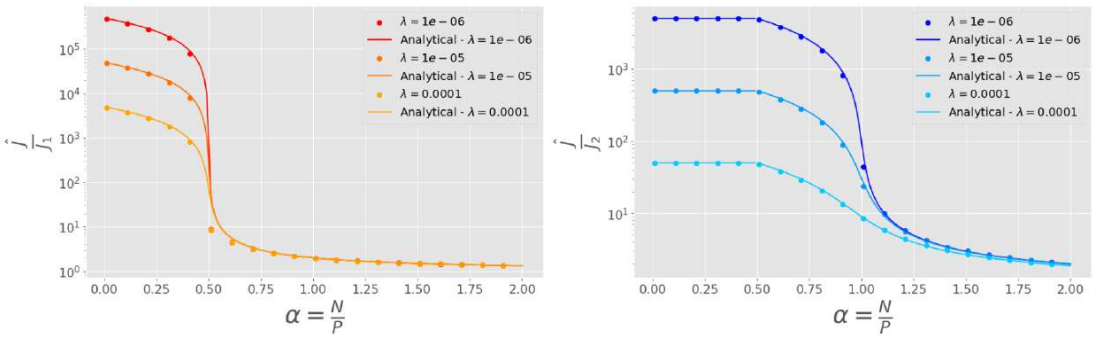

FIGURE 3.6: On the left (A): the *high-variance* eigenvalues are learned in $\alpha \in [0, 0.5]$. On the right (B) the *low-variance* eigenvalues instead are frozen in this interval and are learned for $\alpha > 0.5$. The simulation parameters are $P = 100$, $J_1 = 1$ and $J_2 = 100$. The two modes are divided equally, meaning that $f_1 = f_2 = 0.5$.

$$\begin{aligned} \mathrm{Var}[\hat{J}_1] &= g_1'(-2\lambda) - g_1(-2\lambda)^2 \\ \mathrm{Var}[\hat{J}_2] &= g_2'(-2\lambda) - g_2(-2\lambda)^2 \end{aligned} \tag{3.13}$$

As it can be seen in Figure 3.7 the variance of the first mode is peaking for $\alpha_1^* = \frac{f_1}{2}$ while the variance of the second mode is peaking for $\alpha_2^* = \frac{f_2}{2}$. This is the straightforward generalization of what was found in Equation 3.5 by simply considering this inference task as two separate tasks of smaller size being carried out in sequence.

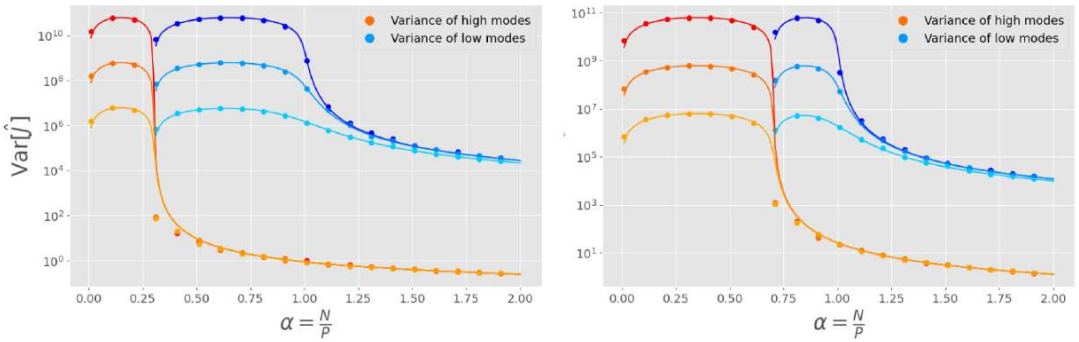

FIGURE 3.7: Variance of the eigenvalues of $\hat{J}$ for two different partitions of the two modes. On the left $f_1 = 0.3$ and $f_2 = 0.7$, while on the right $f_1 = 0.7$ and $f_2 = 0.3$.

In Figure 3.8 we can observe that the critical values of $\alpha_{1,2}^*$ found above correspond to the points of maximal variance *within* the blocks corresponding respectively to high and low variance, as predicted above.

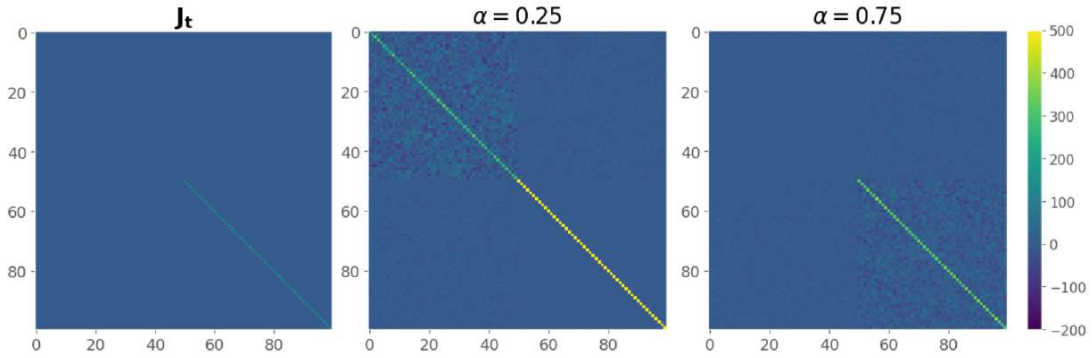

FIGURE 3.8: Inferred coupling matrix $\hat{\mathbf{J}}$ for the values α. Numerical parameters are $P = 100$, $J_1 = 1$, $J_2 = 100$, $f_1 = f_2 = 0.5$. The top half of $\mathbf{J_t}$ is associated to high variance, while the second half is associated to low variance.

Concluding this section, we have proved that in the undersampled regime features are learned in a hierarchical and independent way when the scales involved are widely separated. More specifically, the learning curves above correspond to the curves of two separate inference problems of smaller size being performed from highest to lowest variance. It should be noted that this result can be easily extend to a scenario in which there are not just two, but d-different modes along the diagonal.

3.0.3 The intermediate regime

In subsection 3.0.1 we have described the theory of learning an isotropic model, while in subsection 3.0.2 we have generalized our findings to the case of models with d widely separated scales. Of course we expect that the the prediction of the latter scenario will match that of the former when the separation of scales is reduced. In order to investigate this systematically we can define the *aspect ratio* $\gamma = \frac{J_2}{J_1}$ and we can tune it to observe the transition between the two limits.

As it can be seen in Figure 3.9, when $\gamma = 1$ we retrieve the single peak at $\alpha = 0.5$ predicted by the isotropic theory (subsection 3.0.1). When γ is increased, instead, the curves associated to the two modes start diverging, until when, for $\gamma \gg 1$ they settle on the values of $\alpha^*_{1,2} = \frac{f_{1,2}}{2}$ predicted by the two-scales theory (subsection 3.0.2).

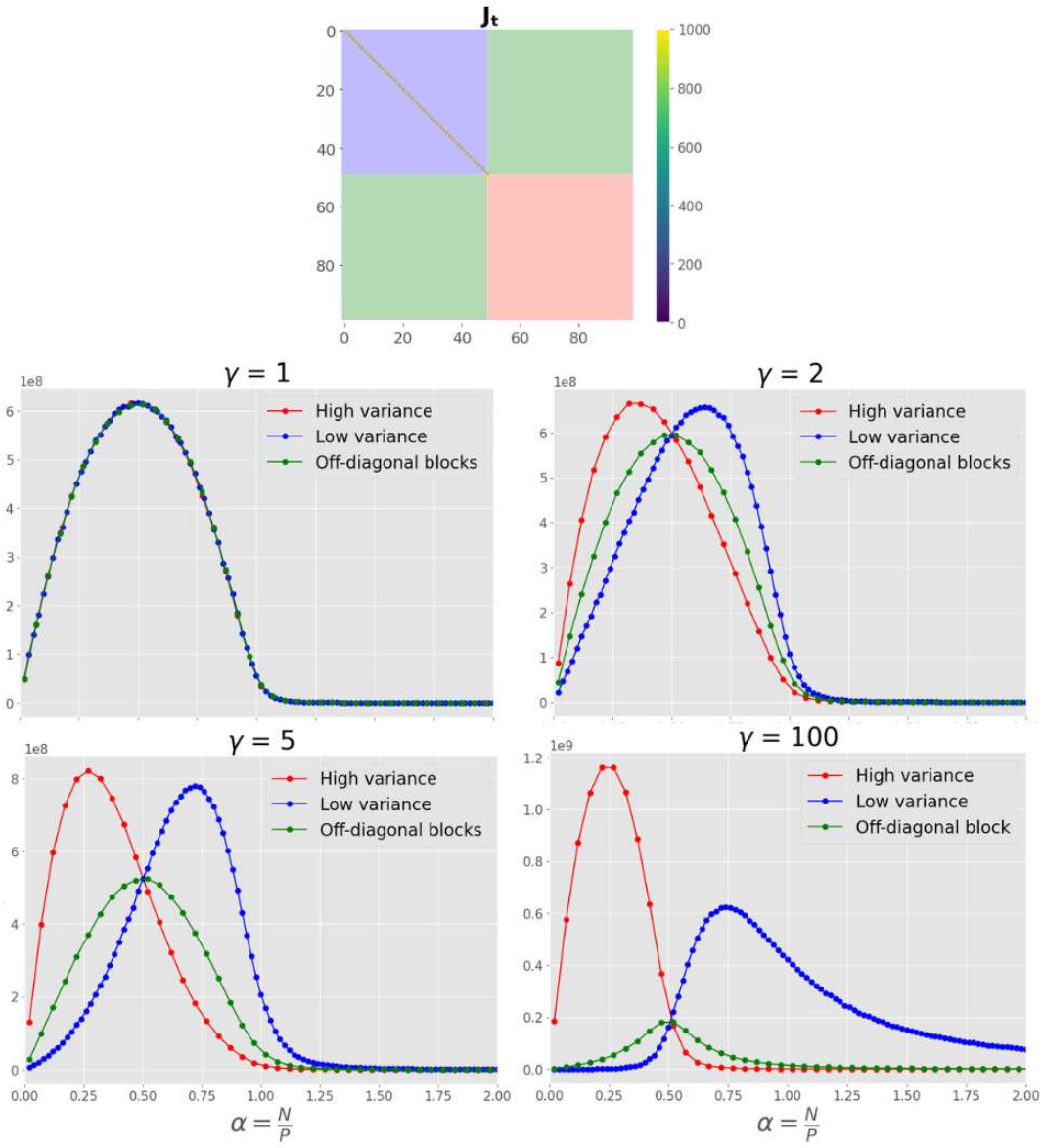

FIGURE 3.9: Variance of the matrix elements of $\hat{\mathbf{J}}$ changing the value of $\gamma = \frac{J_2}{J_1}$. Simulation parameters are $f_1 = f_2 = 0.5$, $P = 100$, $\lambda = 10^{-6}$. The low variance block is represented in blue, high variance is represented in red, and the off-diagonal blocks in green.

Chapter 4

Conclusions

In this work we have discussed the effects of undersampling and structure in the input data of generative models. We first simplified the problem by reformulating it in a Gaussian scenario where a ground truth model is available, and then studied the statistical properties of the inferred parameters through the means of Random Matrix Theory. Notably, we found that even in the simple case of isotropic input data, peaks emerge in the fluctuations of the inferred parameters, for which we provided analytical formulae. When structure is added, multiple peaks are observed, and we provided the criteria needed to understand their position and sharpness, connecting to previous works on Potts models and ridge regression. Additionally, we explain that when structure is added, models learn features in a hierarchical way as a function of the dataset size, going from high to low variance.

This work leaves room for many possible extensions:

- An extensive discussion of models with interactions is needed. In other words, what happens when the teacher model has non-zero off-diagonal elements? The first reasonable task one can tackle is that of fully characterizing the inference of uniform blocks of different sizes and coupling strength along the diagonal, that is developing a theory for the inference of sectors. In this case one can interpret sectors as *signals* to be inferred in presence of the *noise* dictated by the diagonal elements of the covariance matrix.

- The *learning dynamics* of models contain a lot of information about data that is disregarded when working only with maximum-likelihood estimates. The same type of hierarchy that we have unveiled as a function of the dataset size is indeed present as a function of the number of gradient descent iterations needed for learning. This is aspect deserves further exploration and needs to be related to the phenomena described in this work.

- In a scenario in which no ground truth model is available, it is necessary to find some measurable quantity (like a *susceptibility*) during learning that will probe the progressive learning of new modes, in order to provide criteria for understanding *which features* can be learned for a given level of undersampling.

Appendix A

Independence on regularization scheme

In this section, we extend the results obtained for $\tilde{L}_1$ regularization to the more common choice of L_2 regularization. The Max-Likelihood is now

$$\log \mathcal{L}(\mathbf{J} \mid \mathbf{X}) = \frac{1}{2} \log \det(\mathbf{J}) - \frac{1}{2} \operatorname{Tr}[\mathbf{JS}] - \lambda \|\mathbf{J}\|_F^2 \tag{A.1}$$

and we can exploit the fact that $\mathbf{J}$ is symmetric and positive definite (SPD) to obtain that

$$\|\mathbf{J}\|_F^2 = \operatorname{Tr}\left(\mathbf{J}^T\mathbf{J}\right) = \operatorname{Tr}\left(\mathbf{J}^2\right) = \sum_{i,j} J_{ij}^2 = \sum_{i=1}^{n} J_i^2$$

where J_i is the i-th eigenvalue of $\mathbf{J}$. We can find that the eigenvalue problem with this regularization choice is the same as for $\tilde{L}_1$ but we have to be careful and map $\lambda \to \sqrt{\lambda}$.

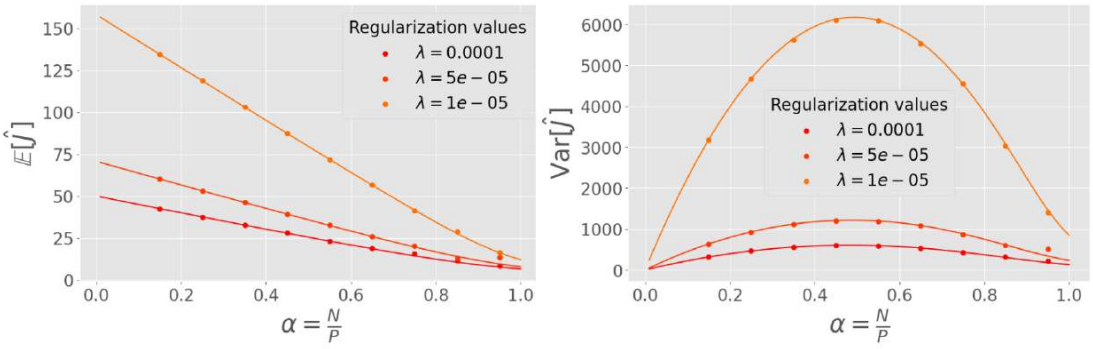

FIGURE A.1: Mean and variance of the eigenvalues of $\hat{\mathbf{J}}$ using L_2 regularization in the context of an isotropic teacher model. Model parameters are $P = 20$, $J_0 = 1$, while gradient descent parameters are learning rate $\gamma = 10^{-2}$ and number of iterations $N = 5000$. Scatter points are simulation data and the solid lines are the theoretical curves found in subsection 3.0.1 by mapping λ to $\sqrt{\lambda}$.

It should be noted that in the case of L_2 regularization we do not have access to an

explicit form of $\hat{\mathbf{J}}$ as we did for $\tilde{L}_1$. therefore we need to run gradient descent on the cost function

$$\log \mathcal{L}(\mathbf{J} \mid \mathbf{X}) = \frac{1}{2} \log \det(\mathbf{J}) - \frac{1}{2} \operatorname{Tr}[\mathbf{JS}] - \lambda \operatorname{Tr}[\mathbf{J^2}] \tag{A.2}$$

which is a delicate process as we have to ensure that $\hat{\mathbf{J}}$ is SPD at each iteration. In order to solve this issue we ran the optimization procedure only on the lower-triangular part of $\mathbf{J}$, which we call $\hat{\mathbf{L}}$, only computing $\hat{\mathbf{J}} = \hat{\mathbf{L}}^{\mathbf{T}}\hat{\mathbf{L}}$ at the end of the procedure.

Appendix B

Bridging with pre existent work

In this section we aim at substantiating the connection between our results and pre-existent works in literature, with a particular focus on the work by Kleeorin et al., 2023. In order to do this, there are three criticalities that need to be addressed:

1. Potts and Gaussian models are fundamentally different. How to bridge the two formalisms?
2. Previous works (Kleeorin et al., 2023) have highlighted the uneven representation of different *structures* in the input data, but in this work we consider *variance modes* as central the central objects of our study. How are these two things related?
3. We argued in subsection 3.0.1 that the peaks in parameter inference observed in Kleeorin et al., 2023 can be explained by looking at the fluctuations of the inferred parameters. Why is that?

In the following we will address all three issues above.

B.1 Gaussian models as Mean Field Potts models

The *Mean Field* approximation (MF) is a powerful technique from statistical physics that allows for a simplified treatment of many systems. The main idea of Mean Field is to assume that each unit of the considered system is subject to an 'average' interaction with all its neighbors, instead of a site-specific interaction.

In the work by Morcos et al., 2011 it was shown that a MF description of a Potts model is available and can be achieved thanks to a low-coupling expansion, originally developed by Plefka in the context of disordered systems. The bottomline of this work is that estimating the parameters of the Potts model within MF reduces to the inversion of a regularized empirical covariance matrix

$$C_{ij}^{(emp)}(A,B) = f_{ij}(A,B) - f_i(A)f_j(B) \tag{B.1}$$

where

$$f_i(A) = \frac{1}{\lambda + M_{eff}} \left(\frac{\lambda}{q} + \sum_{a=1}^{M} \frac{1}{m^a} \delta_{A,A_i^a} \right)$$
$$f_{ij}(A,B) = \frac{1}{\lambda + M_{eff}} \left(\frac{\lambda}{q^2} + \sum_{a=1}^{M} \frac{1}{m^a} \delta_{A,A_i^a} \delta_{B,A_j^a} \right)$$

are the empirical frequencies with the addition of two regularizing elements:

- the *pseudocounts* parameter λ which plays a role similar to regularization,
- a re-weighting by a factor m^a, counting the number of times a sequence is repeated, in order to correct the sampling bias. We also define $M_{\text{eff}} = \sum_{a=1}^{M} 1/m^a$.

It was later shown by Baldassi et al., 2014 that the same matrix as Equation B.1 naturally arises as the covariance matrix of a Gaussian model in a Bayesian inference framework. All in all, the bottomline is that **Gaussian models can be interpreted as the Mean Field version of Potts models**.

B.2 Structure and variance modes

In this work we have used the *variance modes* of the input data as the central object regulating the hierarchy of learning in the undersampled regime. Still, previous works on DCA (see Kleeorin et al., 2023) revolve around the inference of different *features* in the coupling matrix (contacts, sectors, ...) rather than variance modes. In this section we present an argument to show how the two are connected, and how the theory we have proposed in the main body can explain the uneven representation of different features.

Let us consider the true model $\mathbf{J_t}$ as shown in Figure B.1. This 20×20 coupling matrix includes three *contacts*, a *small sector* and a *big sector* as done in Kleeorin et al., 2023 [1]. Let us use the spectral decomposition to write

$$\mathbf{J_t} = \mathbf{ODO}^\top \tag{B.2}$$

where $\mathbf{O}$ is the orthogonal matrix of the eigenvectors of $\mathbf{J_t}$ and $\mathbf{D}$ is the diagonal matrix of its eigenvalues. As it can be seen in Figure B.1, $\mathbf{D}$ presents five different modes along its diagonal, and interestingly the most populated modes are the highest variance ones.

In the main body of this work we have shown that models are *blind* to low-variance modes when data is scarce, and that the inferred parameters relative to *invisible* modes are flattened to a value that is inversely proportional to the regularization strength λ. We can therefore mimick the effect of undersampling by explicitely assigning the value

[1] It should be noted that it is not possible to use the same *exact* matrix that was used in Kleeorin et al., 2023 because Gaussian models carry the additional constraint of their coupling matrix having to be SPD. Here we add a $+2\mathbb{I}$ to the original coupling matrix to ensure this condition is met.

$\frac{1}{\lambda}$ to the low variance modes of $\mathbf{D}$. This will produce a new "flattened" diagonal matrix $\mathbf{D_f}$ (see Figure B.1). Then we can rotate $\mathbf{D_f}$ back in the canonical basis by using

$$\hat{\mathbf{J}}_\mathbf{f} = \mathbf{O}\mathbf{D}_\mathbf{f}\mathbf{O}^\top \tag{B.3}$$

and we see that the *uneven representation of features* emerges in $\mathbf{J_f}$. Indeed, the inferred coupling contact is higher than that of the small sector, which in turn is higher than that of the big sector. Concluding this remark, we have provided an argument to show that the uneven representation of features found in Kleeorin et al., 2023 can be interpreted as the *flattening* of the low-variance modes of the input data, which we have shown to be a consequence of undersampling due to the hierarchical learning scheme we have highlighted in the main body.

B.3 Frobenius norm and fluctuations

When working with Potts models, the couplings are stored in a $q \times q \times L \times L$ tensor $J_{ij}(A, B)$. Still, In order to gain an intuitive understanding of these couplings, it is often desirable to *flatten* this tensor onto a contact matrix. The most common way to do this (as its done in Kleeorin et al., 2023) is to consider the *Frobenius norm* over the amino acids alphabet

$$\left\|J_{ij}\right\| = \left(\sum_{a,b} J_{ij}(a,b)^2\right)^{1/2}.$$

which makes it possible to obtain one simple scalar value for all possible choices of (i, j). We observe that this choice makes it so that $\left\|J_{ij}\right\|$ is a measurement of the fluctuations of the predicted value of J rather then a prediction of the value itself. Consider for example the case of two non interacting sites i and j, for which $J_{ij}(a, b) = 0\ \forall a, b$. In this case $\left\|J_{ij}\right\|$ is proportional to the standard deviation of the coupling strength over all amino acids for sites i and j.

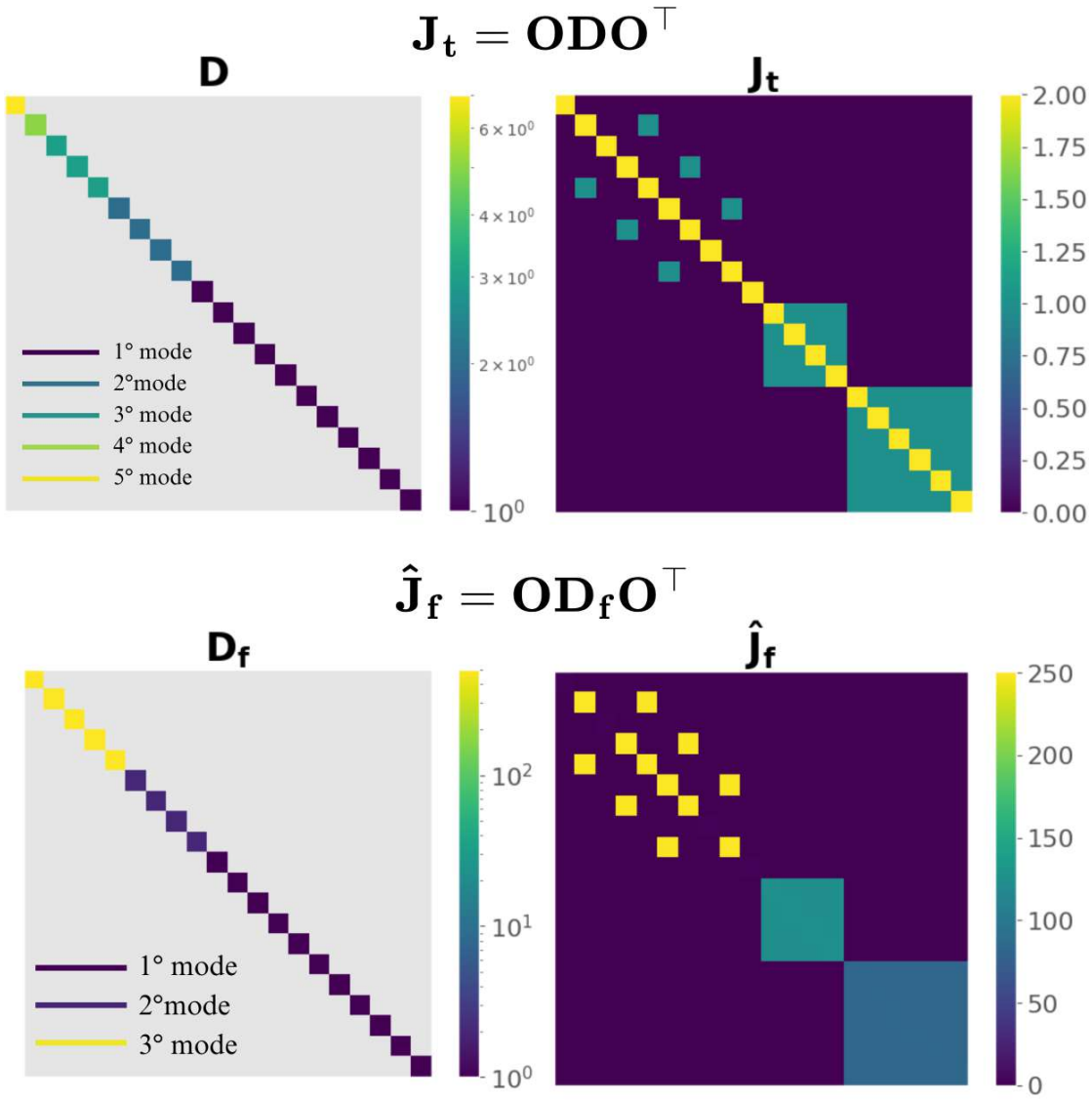

FIGURE B.1: Uneven representation of features emerges when the spectrum of $\mathbf{J_t}$ is flattened in correspondence of the low variance modes. Regularization strength is set to $\lambda = 10^{-3}$.

Bibliography

Baldassi, Carlo et al. (2014). "Fast and Accurate Multivariate Gaussian Modeling of Protein Families: Predicting Residue Contacts and Protein-Interaction Partners". In: *PLoS ONE* 9.3, e92721. DOI: 10.1371/journal.pone.0092721.

Catania, Giovanni et al. (2025). "A theoretical framework for overfitting in energy-based modeling". In: *arXiv preprint arXiv:2501.19158*. URL: https://arxiv.org/abs/2501.19158.

Cocco, Simona et al. (2018). "Inverse statistical physics of protein sequences: a key issues review". In: *Reports on Progress in Physics* 81.3, p. 032601. DOI: 10.1088/1361-6633/aa9965.

Halabi, Najeeb et al. (2009). "Protein sectors: evolutionary units of three-dimensional structure". In: *Cell* 138.4, pp. 774–786. DOI: 10.1016/j.cell.2009.07.038.

Kleeorin, Yaakov et al. (2023). "Undersampling and the inference of coevolution in proteins". In: *Cell Systems* 14.3, 210–219.e7. DOI: 10.1016/j.cels.2022.12.013. URL: https://pubmed.ncbi.nlm.nih.gov/36693377/.

Mel, Gabriel and Surya Ganguli (2021). "A theory of high dimensional regression with arbitrary correlations between input features and target functions: sample complexity, multiple descent curves and a hierarchy of phase transitions". In: *Proceedings of the 38th International Conference on Machine Learning*. Vol. 139. PMLR, pp. 7578–7587. URL: https://proceedings.mlr.press/v139/mel21a.html.

Morcos, Faruck et al. (2011). "Direct-coupling analysis of residue coevolution captures native contacts across many protein families". In: *Proceedings of the National Academy of Sciences* 108.49, E1293–E1301. DOI: 10.1073/pnas.1111471108.

Potters, Marc and Jean-Philippe Bouchaud (2021). *A First Course in Random Matrix Theory*. Cambridge University Press. ISBN: 9781108488082. URL: https://www.cambridge.org/core/books/first-course-in-random-matrix-theory/2292A554A9BB9E2A4697C35BCE920304.

Rocks, Jason W. and Pankaj Mehta (2022). "Memorizing without overfitting: Bias, variance, and interpolation in overparameterized models". In: *Physical Review Research* 4.1, p. 013201. DOI: 10.1103/PhysRevResearch.4.013201. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.4.013201.

Russ, William P. et al. (2020). "An evolution-based model for designing chorismate mutase enzymes". In: *Science* 369.6502, pp. 440–445. DOI: 10.1126/science.aba3304. URL: https://www.science.org/doi/10.1126/science.aba3304.