Generative models for protein sequences

Marion Chauveau

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : ISCE - Ingénierie pour la Santé la Cognition et l'Environnement
Spécialité : MBS - Modèles, méthodes et algorithmes en biologie, santé et environnement
Unité de recherche : Translational Innovation in Medicine and Complexity

Modèles génératifs pour les séquences de protéines

Generative models for protein sequences

Présentée par :

Marion CHAUVEAU

Direction de thèse :

Ivan JUNIER DIRECTEUR DE RECHERCHE, CNRS DELEGATION ALPES	Directeur de thèse
Olivier RIVOIRE DIRECTEUR DE RECHERCHE, CNRS DELEGATION PARIS CENTRE	Co-directeur de thèse

Rapporteurs :

ANNE-FLORENCE BITBOL
ASSISTANT PROFESSOR, ECOLE POLYTECHNIQUE FEDERALE DE LAUSANNE
CYRIL FURTLEHNER
CHARGE DE RECHERCHE HDR, CENTRE INRIA DE SACLAY

Thèse soutenue publiquement le **1er octobre 2025**, devant le jury composé de :

THIERRY MORA, DIRECTEUR DE RECHERCHE, CNRS DELEGATION PARIS CENTRE	Président
IVAN JUNIER, DIRECTEUR DE RECHERCHE, CNRS DELEGATION ALPES	Directeur de thèse
OLIVIER RIVOIRE, DIRECTEUR DE RECHERCHE, CNRS DELEGATION PARIS CENTRE	Co-directeur de thèse
ANNE-FLORENCE BITBOL, ASSISTANT PROFESSOR, ECOLE POLYTECHNIQUE FEDERALE DE LAUSANNE	Rapporteure
CYRIL FURTLEHNER, CHARGE DE RECHERCHE HDR, CENTRE INRIA DE SACLAY	Rapporteur
MARCO RIBEZZI, SENIOR SCIENTIST, QUANTUM-SI FRANCE	Examinateur

Es steht alles schon bei Dedekind.
(It is already all in Dedekind.)

— Emmy Noether

Contents

Acknowledgments

En premier lieu, je souhaite remercier mes directeurs de thèse, Olivier et Ivan. Merci pour votre accompagnement, tant sur le plan scientifique qu'humain, pour la confiance que vous m'avez accordée, ainsi que pour vos précieux conseils. Je mesure la chance que j'ai eue de travailler avec vous.

Je remercie les membres de mon jury, pour avoir accepté l'invitation à participer à cette soutenance et pour le temps qu'ils ont consacré à la lecture du manuscrit: Anne-Florence Bitbol, Cyril Furtlehner, Thierry Mora et Marco Ribezzi.

J'aimerais également exprimer ma gratitude à Martin Weigt et Laurent Oudre pour avoir accepté de faire partie de mon comité de suivi de thèse. Laurent, tu as aussi été un maître de stage exceptionnel, et le premier avec qui j'ai eu la chance de publier un article. Merci pour ton accompagnement et pour tout ce que cette expérience m'a appris.

During this PhD, I had the opportunity to collaborate with many students and researchers, whom I would like to thank here.

Many thanks to Yaakov Kleeorin, Emily Hinds, and Rama Ranganathan for their trust and for their valuable contributions to the work presented in this thesis. I am also deeply grateful to all members of the Ranganathan Lab for their warm welcome during my stay at the University of Chicago. I would like to extend a special note of thanks to Eric and Fatima for taking me up to the rooftops of Chicago and for introducing me to deep-dish pizza on the beach.

I would also like to thank Paul Guenon, Damien Laage, Guillaume Stirnemann, Karolina Filipowska, Kim Reynolds, Clément Nizak, Amaury Paveyranne, Timothé Lucas, and Shoichi Yip for fruitful collaborations and enriching discussions during my thesis.

Au-delà des collaborations scientifiques, ces années de thèse ont aussi été rythmées par des lieux, des bureaux, et plus encore par celles et ceux qui les ont habités.

Le premier d'entre eux, le bureau en semi sous-sol du Collège de France, a une importance particulière à mes yeux. C'est là que tout a commencé, et que j'ai appris à connaître mes futurs acolytes de thèse. Angelo, Amaury, je suis tellement reconnaissant d'avoir mené cette thèse à vos côtés. Sans vous cette aventure aurait été évidemment moins drôle, mais aussi infiniment moins riche humainement et scientifiquement. Merci pour tout, vous êtes des chefs.

Le bureau du bout du monde, à Gulliver, où j'ai passé de très beaux moments jusqu'au déménagement, avec un équipage composé de Yann, Charles, Natalie, Benjamin, Paul et Michel. Puis le bureau B160, où j'ai eu le plaisir de côtoyer chaque jour des personnes bienveillantes et inspirantes : Armand, Coline, Yorgos, Léo, Timothé, Milo, Simone, Barnabé et Romain. J'aimerais en profiter pour remercier l'ensemble du laboratoire Gulliver. La bienveillance et la richesse scientifique de cet environnement ont beaucoup contribué à faire de cette thèse une expérience à la fois sereine et stimulante.

Le bureau Grenoblois, au milieu des montagnes. Merci aux membres de l'équipe Tree pour leur accueil chaleureux, et plus particulièrement à ceux de l'équipe CompBio. Les retraites en montagne ont été des parenthèses privilégiées, à la fois conviviales et inspirantes. Une pensée toute particulière pour Flora et Moira, qui m'ont accueilli chez elles chaque fois que j'en ai eu besoin.

Comment ne pas mentionner le bureau 414 du LJP, dans lequel j'ai souvent trouvé refuge, en squatteur assumé. Merci à ses occupants: Guillaume, Romain, Martin, Alexandre, Angelo, Amaury, ainsi qu'aux autres squatteurs de ce bureau: Mélicia, Thibault, Antoine et Louise sans qui ce manuscrit contiendrait sans doute encore bien des fautes. À vous tous, j'ai envie de dire combien j'aime ce que vous êtes, et ce que je suis à vos côtés.

Impossible de clore ces remerciements sans un mot pour les P'tits Lapins. Une pensée particulière pour Nicolas et Yoann, et leur aide souvent précieuse en informatique (sans oublier cette victoire mémorable au TRACS). Simon et Zoé, pour cette coloc inoubliable. À vous tous, merci d'être encore là dix ans après. Les p'tits lapins c'est mieux qu'un groupe, une famille, une bande, une tribu, c'est tout ça à la fois.

Enfin, merci à ma famille,
À mon frère, à toi Bastien,
À ma mère, qui m'a transmis le goût de lire,
À mon père, l'œil du tigre, toujours.

Abstract

Proteins are central to most cellular functions. Elucidating the relationship between the amino acid sequence, the three-dimensional structure, and the biological function is therefore a key challenge in biology.

The first part of the thesis focuses on generative models for protein sequences. Given a multiple sequence alignment (MSA) of proteins with similar structures and functions, the objective is to develop a model that can generate novel sequences capturing the diversity and functional properties of natural ones.

In particular, we are interested in Potts models trained to capture single-site and pairwise amino acid frequencies within an MSA. These models, also called bmDCA for Bolztmann Machine Direct Coupling Analysis, have been successfully applied to tasks such as predicting protein contacts, evaluating mutational effects, and designing functional protein sequences.

Despite their successes, bmDCA models face a major limitation: the size of the MSA is typically too small relative to the model's complexity. Without regularization, this imbalance severely limits their ability to generalize beyond the observed data.

The standard L_2 regularization commonly used in this context introduces its own challenges: (i) generating functional sequences requires modifying the sampling procedure, which in turn reduces the diversity of synthetic sequences; (ii) the different statistical features observed in the sequence data are captured with unequal accuracy.

To address these issues, we propose a new method called Stochastic Boltzmann Machine (SBM), which relies on a Quasi-Newton gradient descent with implicit regularization. We evaluate our approach on synthetic data and show its efficacy in correcting uneven representation of different statistical patterns. In collaboration with experimentalists from the University of Chicago, we show that functional protein sequences can be generated without modifying the sampling procedure, thereby preserving the diversity of the synthetic dataset. Finally, we discuss the persistent difficulty of evaluating such generative models, even when they reproduce empirical statistics accurately.

In the second part of the thesis, we explore the ability of the same models to predict compensatory mutations in protein sequences. Specifically, we introduce a method to identify mutations that exhibit positive epistatic interactions.

We first apply this approach to serine proteases, a family of enzymes known for their functional diversity. The mechanisms underlying this diversity, and the ability to switch protein function through a limited number of mutations, have been a major focus of investigation. Starting from a well-chosen mutation, we aim to predict compensatory mutations that could collectively shift the function. We compare our predictions with results from the literature and provide preliminary experimental validation, obtained by collaborators at Sorbonne University.

Finally, we investigate the case of *E. coli* dihydrofolate reductase, in which a mutation distant from the active site is known to severely reduce catalytic activity. This long-range effect suggests an underlying allosteric mechanism, for which collaborators at ENS have proposed a description using molecular dynamics simulations. Using our method, we predict a compensatory mutation that further supports the understanding of this allosteric mechanism.

Résumé

Les protéines jouent un rôle central dans la majorité des fonctions cellulaires. Comprendre le lien entre la séquence d'acides aminés, la structure tridimensionnelle et la fonction biologique constitue donc un enjeu majeur en biologie.

La première partie de cette thèse s'intéresse aux modèles génératifs pour les séquences de protéines. À partir d'un alignement multiple de séquences ayant des structures et des fonctions similaires, l'objectif est de construire un modèle capable de générer de nouvelles séquences reproduisant la diversité et les propriétés fonctionnelles des protéines naturelles.

Nous nous intéressons en particulier aux modèles de Potts, entraînés à reproduire les fréquences individuelles et les fréquences des paires d'acides aminés au sein d'un alignement. Ces modèles, également appelés bmDCA (Boltzmann Machine Direct Coupling Analysis), ont démontré leur efficacité dans des tâches telles que la prédiction des contacts dans la structure tridimensionnelle, l'évaluation des effets des mutations ou encore la génération de séquences de protéines fonctionnelles.

Malgré leurs succès, les modèles bmDCA souffrent d'une limite majeure : la taille de l'alignement est généralement trop faible par rapport à la complexité du modèle. En l'absence de régularisation, ce déséquilibre restreint fortement leur capacité à généraliser au-delà des données observées.

La régularisation L_2, couramment utilisée dans ce contexte, soulève certaines difficultés : (i) générer des séquences fonctionnelles nécessite de modifier la procédure d'échantillonnage, ce qui réduit la diversité des séquences synthétiques ; (ii) les différentes caractéristiques statistiques observées dans les données de séquences ne sont pas capturées de manière équivalente par le modèle.

Pour répondre à ces enjeux, nous proposons une nouvelle méthode, la Stochastic Boltzmann Machine (SBM), qui repose sur une descente de gradient quasi-Newton intégrant une régularisation implicite. Nous évaluons cette approche sur des séquences synthétiques pour montrer qu'elle permet de mieux capturer les différents motifs statistiques présents dans les données. En collaboration avec des expérimentateurs à l'université de Chicago, nous montrons qu'il est également possible de générer des séquences fonctionnelles sans modifier la procédure d'échantillonnage, ce qui permet de préserver la diversité du jeu de données artificiel. Enfin, nous discutons de la difficulté à évaluer ces modèles génératifs, même lorsqu'ils reproduisent fidèlement les statistiques empiriques.

Dans la seconde partie de la thèse, nous explorons la capacité de ces mêmes modèles à prédire des mutations compensatrices dans les séquences de protéines. Plus précisément, nous introduisons une méthode permettant d'identifier des mutations ayant des interactions épistatiques positives.

Nous appliquons d'abord cette approche à la famille des protéases à sérine, connue pour sa grande diversité fonctionnelle. Les mécanismes à l'origine de cette diversité, et notamment la possibilité de modifier la fonction d'une protéine avec peu de mutations, ont fait l'objet de nombreuses études. À partir d'une mutation ciblée, nous cherchons à prédire des mutations compensatrices susceptibles d'entraîner un changement de fonction. Nous confrontons nos prédictions aux données de la littérature et présentons des résultats expérimentaux préliminaires obtenus par des collaborateurs de l'université de Sorbonne.

Enfin, nous nous intéressons à la dihydrofolate réductase d'*E. coli*, dans laquelle une mutation distante du site actif est connue pour réduire fortement l'activité catalytique. Cet effet à longue portée suggère l'existence d'un mécanisme d'allostérie, pour lequel des collaborateurs à l'ENS ont proposé une description grâce à des simulations de dynamique moléculaire. Dans ce contexte, nous utilisons notre méthode pour prédire une mutation compensatrice venant appuyer la compréhension de ce mécanisme allostérique.

Chapter summaries & research contributions

Part I: Introduction to generative protein sequence models

Chapter 1 provides a brief introduction to protein properties and key concepts that will be used and referred to throughout the manuscript.

Chapter 2 provides a deeper exploration of statistical inference applied to protein sequences, with particular emphasis on the Boltzmann Machine (BM), which serves as the modeling framework in this thesis. This chapter also highlights key challenges associated with generative models, such as the evaluation of their performance.

Part II: The undersampling problem

Chapter 3 addresses the issue of undersampling in Boltzmann Machine models. We take a closer look at two important results from the literature: (1) the biases identified in BM inference in the context of undersampled data, and (2) the generative power of BM models, as tested in 2020 on the Chorismate Mutase family.

Chapter 4 presents the first main contribution of this thesis: a novel inference method, the Stochastic Boltzmann Machine, designed to mitigate undersampling biases in generative modeling of protein sequences. At the time of writing, this work has not yet been published but has been presented in the form of contributed talks or posters at scientific conferences. A manuscript is currently in preparation.

Part III: Beyond the distribution of natural protein sequences

Chapter 5 is an introductory chapter that outlines key concepts related to protein properties, functions, and mutational effects. Particular emphasis is placed on the notion of epistasis and several results from the literature that illustrate the complexity of residue interactions in protein sequences. We also explain how mutational effects can be measured at scale using Deep Mutational Scanning (DMS) methods, and how statistical models leverage protein sequence data to predict these effects.

The next two chapters present the second contribution of the thesis, namely the use of Boltzmann Machine models to guide the prediction of compensatory mutations in protein sequences. In particular, we focus on two specific objectives: (i) investigate the determinants of specificity within the serine protease family, as detailed in **Chapter 6**, and (ii) support the understanding of an allosteric mechanism in the dihydrofolate reductase family, as explored in **Chapter 7**.

As for Chapter 3, this work has not been shared through a publication, but has been presented at scientific conferences. Notably, the results discussed in Chapter 7 are part of a scientific article currently in preparation:

> [1] Paul Guenon, **Marion Chauveau**, Karolina Filipowska, Clément Nizak, Guillaume Stirnemann, Kimberly Reynolds, Olivier Rivoire, Damien Laage. *Allostery without Large Motions: Molecular Dissection of a Minimal-Shift MWC Allosteric Regulation*. *In preparation*.

Finally, **Chapter 8** concludes the manuscript by summarizing the methods explored and the key findings of this thesis.

Other works

During the third year of my PhD, I contributed to the development of a Python library for coevolution analysis on protein sequence data. This work is the subject of a forthcoming publication:

> [2] Margaux Julien, **Marion Chauveau**, William Schmitt, Fabien Pierrel, Sophie Abby, Nelle Varoquaux, Ivan Junier. *COCOA-Tree: COllaborative COevolution Analysis Toolbox. In preparation*.

Beyond the scope of this thesis, I have also been involved in the following research projects—carried out in the continuation of my master's internship at the Centre Borelli—which are not further discussed in this manuscript:

> **Marion Chauveau**, Antoine Mazarguil, Laurent Oudre. *Graph dictionary learning for the study of human motion*. In *Proceedings of the 2024 46th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1–5, 2024. IEEE.

> Sylvain W. Combettes, Paul Boniol, Antoine Mazarguil, Danping Wang, Diego Vaquero-Ramos, **Marion Chauveau**, Laurent Oudre, Nicolas Vayatis, Pierre-Paul Vidal, Alexandra Roren, et al. *Arm-CODA: A Data Set of Upper-limb Human Movement During Routine Examination*. *Image Processing On Line*, vol. 14, pp. 1–13, 2024.

Introduction to Generative Protein Sequence Models

Proteins 1

Proteins are the most abundant macromolecules in cells. They not only provide the structural framework that defines cellular shape, but also serve as the main agents of most cellular function. This chapter provides a concise overview of essential protein characteristics, outlines key challenges in their study, and highlights some technological breakthroughs that have significantly shaped research in this field.

1.1 DNA to Amino Acid Sequence

The study of DNA has a long and collaborative history, involving many researchers since the mid-19th century. Their combined efforts have led to a detailed understanding of how hereditary information is encoded, stored, and transmitted across generations [3].

In all living organisms, the genetic information is organized into units called genes, which together constitute the genotype. Each gene consists of a sequence of nucleotides, the fundamental building blocks of DNA, composed of four bases: adenine (A), cytosine (C), guanine (G), and thymine (T).

The principle of complementary base pairing, in which specific bases form stable pairs, not only ensures DNA replication but also enables the transcription of genes into messenger Ribonucleic Acid (mRNA). During this process, the DNA sequence of a gene is thus copied into a complementary RNA sequence.

Once produced, the mRNA serves as a template for protein synthesis through a process known as translation, during which molecular machines called ribosomes read the mRNA sequence to assemble a corresponding chain of amino acids.

Proteins produced through this process are composed of one or more polypeptide chains, which are linear sequences of amino acids linked by covalent peptide bonds. As illustrated in Figure 1.1, all amino acids share a common structural framework centered around the C_α atom, which is bonded to an amino group, a carboxyl group, a hydrogen atom, and a variable side chain (denoted as R). It is the nature of this side chain that distinguishes one amino acid from another.

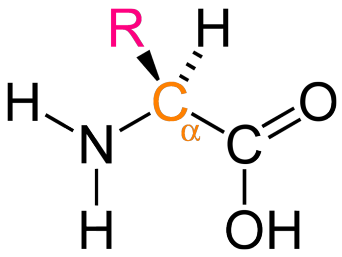

Figure 1.1: General formula of an amino acid. (Figure from [4])
The central carbon of the amino acid, also called C_α is displayed in orange. On the left we have the Amine group, and on the right the Carboxyl group. R represents the side chain of the amino acid.

Figure 1.2 presents the chemical structures of the 20 standard amino acids[1], along with their names and abbreviations that will be used throughout this manuscript. The side chains of these amino acids exhibit a wide range of chemical properties: they may carry a positive or negative charge, be polar or nonpolar, and vary in their affinity for water, from hydrophilic to hydrophobic [5].

1: Two additional amino acids (selenocysteine and pyrrolysine) are found in certain organisms, but will not be considered in this manuscript.

Ultimately, the unique sequence of amino acids, and by extension, the sequence of their chemically diverse side chains, determines the structure and the function of the resulting protein. Once incorporated into a polypeptide chain, an amino acid is typically referred to as a *residue*, a convention we will follow throughout this manuscript.

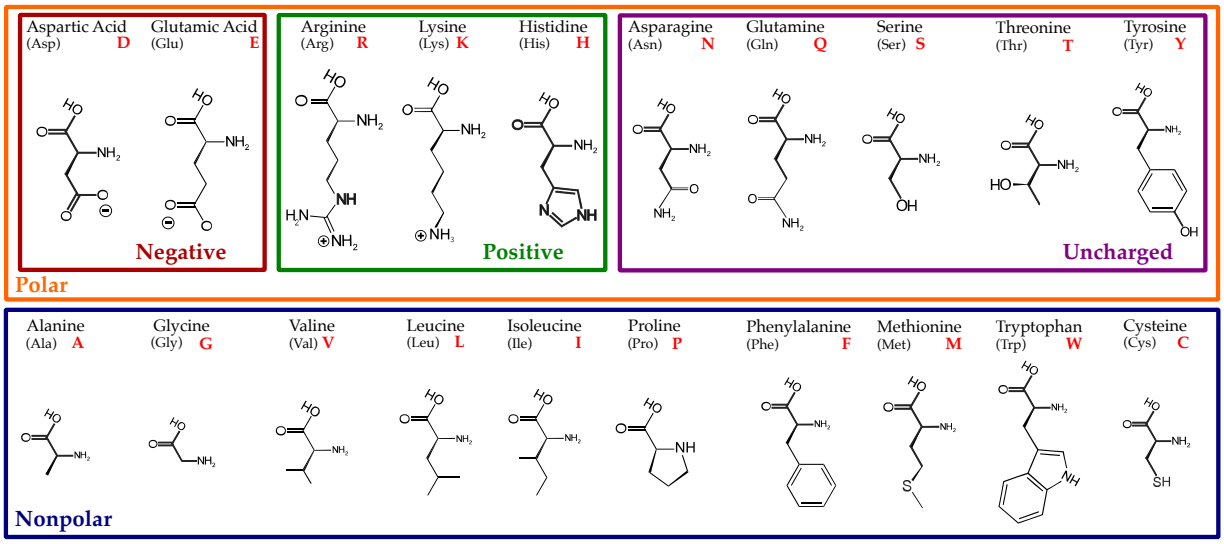

Figure 1.2: The 20 standard amino acids, classified according to the properties of their side chains.
Each amino acid is represented by its full name, three-letter abbreviation, one-letter code (in red), and chemical structure.

1.2 Structure

After its transcription by ribosomes, a protein folds into a state of minimum free energy known as its native conformation and also called tertiary structure [6]. This folded state is primarily determined by the linear sequence of amino acids and the physicochemical conditions of the surrounding solution, such as temperature and pH [7]. However, this conformation is not necessarily static: it can change, especially upon interaction with other molecules, an essential feature for many protein functions [8].

Most proteins are composed of 50 to 2000 amino acids [9], and larger proteins often contain multiple interacting domains, i.e. structurally independent units that each adopt their own tertiary conformation. When several such domains come together, they form what is referred to as the quaternary structure of the protein. Notably, a given domain can often be found in a variety of different proteins.

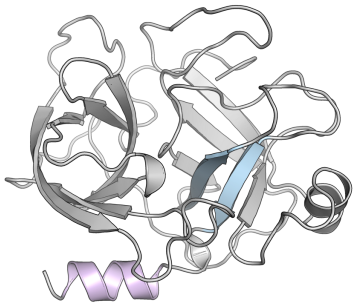

Figure 1.3: α helix and β sheet
Examples of an α helix (pink) and β sheet (blue) highlighted on a ribbon representation of the Rat Trypsin protein (PDB: 3TGI).

To describe protein structures in more detail, it is useful to recognize that many proteins are built from recurring secondary structure motifs, particularly α helices and β sheets. These patterns are especially prevalent because they arise from regular hydrogen bonding between backbone N–H and C=O groups, independent of the side chains. Figure 1.3 depicts these secondary structures, which are represented as spiraling ribbons for α helices and aligned arrows for β sheets.

Given the complexity of protein structures, various visual representations are commonly used to emphasize different features. For example, Figure 1.4 presents standard representations of protein structures that we will use throughout this manuscript (ribbon and space-filling models) all generated using the PyMOL software [10].

All the structures presented in this manuscript originate from the RCSB Protein Data Bank [11] and were obtained through experimental methods. The most commonly used techniques include X-ray crystallography [12], which needs the formation of protein crystals to analyze X-ray diffraction patterns, and nuclear magnetic resonance (NMR) spectroscopy [13],

which can also capture dynamic conformational changes in proteins. However, because these experiments are time-consuming and costly, only 0.05% of the protein sequences in the public database UniProtKB [9] have experimentally determined structures[2].

2: https://www.rcsb.org/statistics/growth/unique-uniprot-entries

Predicting protein structure from amino acid sequence has therefore emerged as a central challenge. The availability of some experimentally determined structures, combined with the abundance of protein sequence data, has provided a strong foundation for training machine learning models capable of accurate structure prediction. In this context, the potential of deep learning became evident with the breakthrough performance of AlphaFold in the 2018 edition of the Critical Assessment of Structure Prediction (CASP) [14].

Structure prediction is a powerful tool for getting a deeper understanding of proteins. However, it represents only one part of the puzzle. Whether obtained experimentally or computationally, protein structures are often static snapshots of conformations, which may differ from the dynamic and context-dependent environments in which proteins operate.

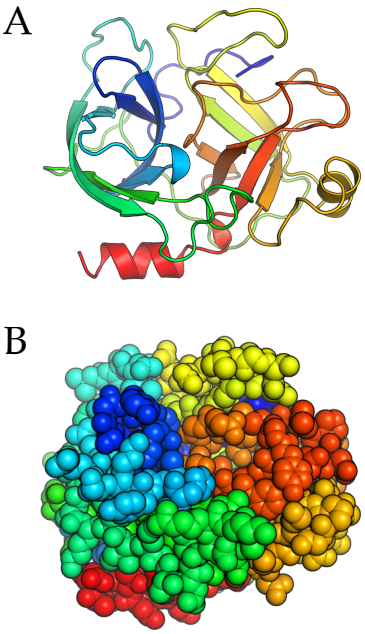

Figure 1.4: Examples of protein structure representation.
A. Ribbon model: highlights the protein's secondary structure elements such as α helices (coils) and β sheets (arrows).
B. Space-filling model: shows the van der Waals surface of the protein, emphasizing the overall shape and volume.

1.3 Function

Proteins perform a wide variety of functions in living organisms. They provide structure, transport molecules, transmit signals, catalyze chemical reactions, protect against pathogens and so on [5]. While this thesis is not dedicated to a specific functional category, all the examples on which we will focus in the following happen to fall under the category of enzymes.

Enzymes are proteins that act as catalysts, which means that they accelerate biochemical reactions without being consumed. These proteins can also be classified into several categories, each highly specialized and effective in catalyzing a particular type of reaction. For example, proteases are enzymes that cleave peptide bonds, a function essential for digestion or cellular regulation[3].

While functional labels such as "enzyme," "receptor," or "transporter" are helpful for studying proteins, it is important to emphasize that all these types of proteins rely on molecular interactions to fulfill their roles. Antibodies attach to pathogen and flag them for elimination; actin monomers assemble into filaments; enzymes bind substrates to accelerate chemical reactions. In the end, it is all these interactions that are at the foundation of cellular life.

3: In Section 5.1.3, we briefly describe the experimental approaches used to evaluate the functional activity of enzymes.

1.3.1 Revolution at the lab bench

Part of the data presented in this thesis was obtained through experiments carried out by collaborators. Such studies are made possible by tools that allows us to manipulate DNA in a test tube or an organism. These include the ability to cut and ligate DNA fragments, enabling the modular assembly of genetic elements; to clone DNA, producing billions of copies from a single molecule [15]; to chemically synthesize arbitrary DNA sequences, even those not found in nature [16]; and to read the exact nucleotide composition of any DNA molecule through sequencing methods [17]. Thanks to these tools, we can, for example, engineer cells to express proteins (artificial or not), while knocking out the gene coding for

the native version, a strategy that will appear at several points throughout this thesis.

One of the most significant revolutions, if not the most significant, in modern biology has been the advent of *next-generation* sequencing technologies, which have led to a dramatic reduction in sequencing costs.[4] The introduction of Illumina platforms in 2006–2007 marked the rise of second-generation sequencing [17], followed by the development of third-generation methods such as Nanopore sequencing in 2015 [18].

4: Today, sequencing a human genome costs on the order of a thousand dollars, compared to approximately 3 billions dollars at the time of the first genome project, before the advent of next-generation approaches.

If understanding what proteins do, and how they do it, has long stood as a central question in biology, our capacity to address it has changed dramatically over the past three decades thanks to these technological advances. Large-scale repositories like UniProtKB now host billions of sequences [9], providing the raw material for novel lines of research. These include approaches grounded in statistical physics, presented in Chapter 2, as well as deep learning methods, briefly reviewed in Section 2.2.

1.3.2 Function through evolution

How is it possible that most proteins in living organisms consistently adopt well-defined, stable conformations? And how do protein interactions with other molecules achieve such remarkable specificity, in that they typically bind only one or a few partners among thousands of potential interactors? The answer lies in evolution through natural selection.

During evolution, several mechanisms generate sequence diversity. Spontaneous point mutations arise from replication errors and environmental damage; gene duplication creates redundant copies that can diverge, yielding *paralogues*; and horizontal gene transfer moves genetic material between cells. The proteins we observe today represent the subset of variants that have survived natural selection, maintaining or improving the organism's capacity for survival, growth, and reproduction in a given environment, collectively referred to as fitness.

Despite sometimes considerable divergence, many of these protein sequences still share a common structure and related functions. Recognising such evolutionary relationships is essential, and is formalised through the concepts of *homology* and *protein families*.

1.4 Protein families & homology

In evolutionary biology, homology refers to the similarity between characteristics of organisms that arises from their shared ancestry. When applied to proteins, it underpins the concept of families: groups of proteins that, despite having diverged over long evolutionary timescales, retain significant structural and functional similarities [20].

An illustrative example is provided by the serine proteases, a large family of enzymes that includes, among others, the digestive enzymes chymotrypsin and trypsin [21]. As shown in Figure 1.5A, the similarity between their three-dimensional structures is remarkable. If we now examine amino acid sequences of proteins belonging to this family, as depicted in Figure 1.5B, we observe regions of strong conservation as well as positions showing considerable variability.

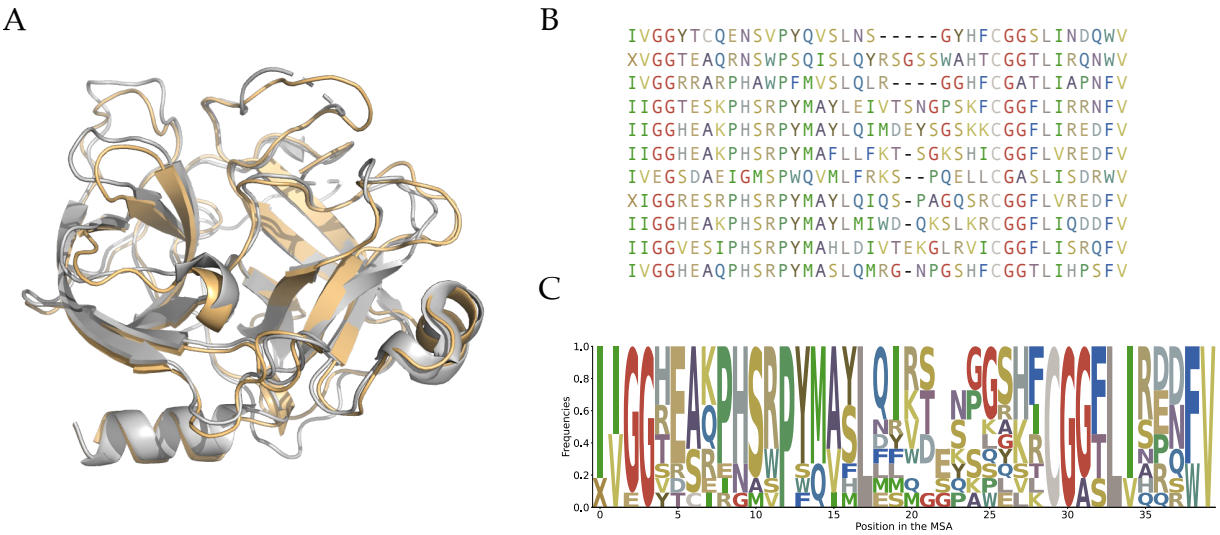

Figure 1.5: Example of the S1A family.
A. Superposition of the structures of rat trypsin (PDB 3TGI, beige) and bovine chymotrypsin (PDB 1T8O, grey), illustrating the high structural similarity between these enzymes.
B. Segment of a multiple sequence alignment (adapted from [19]).
C. Sequence logo representation showing amino acid frequencies at each site of the alignment. For example, position 12 is highly conserved, while position 25 exhibits significant amino acid variability.

Some of these sequence differences were likely favored by natural selection because they, for example, allowed the protein to acquire new functional properties. However, many other amino acid substitutions are neutral, having little to no impact on the overall structure or function of the protein. Remarkably, a relatively small number of conserved amino acids may suffice to preserve the same structural fold and biological function. Thus, it is not uncommon for two proteins that share a fold and function to differ in more than 70% of their amino acid sequences [22]. This does not mean that all non-conserved residues in such proteins are unimportant for structure or function, but rather that their role emerges through interactions with other residues, a point to which we will return in Section 1.5.1.

While this discussion has focused on serine proteases, the same principles apply to hundreds of other protein families; in general, the three-dimensional structure of family members is more highly conserved than their amino acid sequences.

To systematically study protein families, we can rely on resources such as Pfam (now integrated into the Interpro database [23]), which, for example, catalogues proteins based on their family classification. In the context of this thesis, we will typically make use of such databases and particularly the multiple sequence alignments they provide.

1.4.1 Multiple Sequence Alignments

A multiple sequence alignment (MSA) is obtained through a computational method that typically identifies regions of similarity that may indicate functional, structural, or evolutionary relationships. In an MSA, sequences are arranged so that homologous residues are aligned in columns, and gaps are introduced to account for insertions or deletions that have occurred over evolutionary time [24]. At the end of the process, we obtain M sequences, each having the same length L.

An example of such an alignment is shown in the case of serine proteases in Figure 1.5B. MSAs can also be visualized as profile logos, where the height of each letter reflects the frequency of the corresponding amino acid at that position (Figure 1.5C), providing an intuitive representation of sequence conservation and variability.

Several methods exist for constructing MSAs, including widely used tools such as MAFFT [25] and MUSCLE [26], each with its own optimization strategy.

In practice, to obtain an MSA for a given protein family, one can either obtain an alignment from databases such as Pfam, or construct a custom alignment. The latter typically involves the selection of representative sequences based on prior knowledge of well-characterized members, combined with automatic searches across large protein databases such as UniProtKB.[5]

5: The HMMER suite [27] is for example widely used to build a profile Hidden Markov Model (HMM) from a seed alignment. This model can then be used to search databases for additional family members.

Defining what constitutes a "good" alignment, choosing appropriate sequences, defining an objective function, and optimizing it, is a challenging task that remains an active area of research. A detailed discussion of these issues is beyond the scope of this work; the interested reader is referred to relevant reviews [24, 28].

The MSAs used in this thesis are either derived from previous studies, cited where appropriate, or provided by collaborators, as is the case for the alignment used in Chapter 6.

Once an alignment has been obtained, it can serve as a starting point to perform various analyses. Indeed, as proteins have evolved under the pressure of natural selection, their sequences encode rich information about the evolutionary constraints that have shaped their function and structure.

1.5 Statistical analysis of MSAs

Throughout this manuscript, we denote an amino acid sequence m of length L as $\left\{\sigma_i^{(m)}\right\}_{i=1}^{L}$. Given an MSA of size M, we can calculate the empirical frequency $f_i(a)$ of observing amino acid a at position i, as well as the joint frequency $f_{ij}(a,b)$ of observing amino acids a and b at positions i and j respectively[6]:

6: $f_{ij}(a,b) = f_{ji}(b,a)$, but in general $f_{ij}(a,b) \neq f_{ij}(b,a)$

$$f_i(a) = \frac{1}{M}\sum_{m=1}^{M}\delta(\sigma_i^{(m)}, a), \tag{1.1}$$

$$f_{ij}(a,b) = \frac{1}{M}\sum_{m=1}^{M}\delta(\sigma_i^{(m)}, a)\delta(\sigma_j^{(m)}, b) \tag{1.2}$$

where $\delta(x,y) = 1$ if $x = y$, and 0 otherwise.

The frequency profiles offer a quantitative glimpse into the variability and conservation of residues across a protein family. As already mentioned, this information can be visualized through sequence logos as shown in Figure 1.5. It is also possible to quantify how much the amino acid distribution at a given site deviates from an expected neutral drift using the Kullback–Leibler relative entropy [29].

1.5.1 Scales of Coevolution

Beyond conservation, statistical dependencies between positions can reveal patterns of coevolution. As illustrated in Figure 1.6, couplings between amino acids that are crucial for maintaining function or structure typically lead to correlated mutations, or coevolution, which can be detected through statistical analysis.

A basic measure of such statistical dependency is this correlation, defined as the deviation of the joint frequency from the product of the marginals:

$$C_{ij}(a,b) = f_{ij}(a,b) - f_i(a)f_j(b). \tag{1.3}$$

As shown in Figure 1.7, this correlation matrix computed on a given protein family typically spans a wide range of scales, both in terms of intensity and in the number of positions involved.

Numerous studies focusing on different protein families have shown that these correlations are indeed present across multiple scales [30–32].

At the smallest scale, pairwise coevolution typically reflects direct physical interactions, such as atomic contacts in the three-dimensional structure [33–35] (see Figure 1.6).

Beyond these local features, a continuum of collective interactions has also been identified. For instance, some networks of coevolving residues, often referred to as sectors [19, 29, 32, 36], have been linked to key functional properties, including binding affinity, enzymatic activity [19], and allosteric communication [36][7].

However, these multi-scale interactions are not directly inferred from raw statistical correlations. Indeed, interpreting correlations is challenging for several reasons: the data are typically in a regime of undersampling; two positions may appear correlated without direct interaction; and sampling biases arise from the evolutionary relationships between sequences, as well as the overrepresentation of certain species in sequence databases.

These challenges have motivated the development of a wide range of computational approaches aimed at extracting structural and functional insights from sequence data [29, 37, 38].

In Chapter 2, we will delve into the Direct Coupling Analysis (DCA) framework, which has been used both to predict tertiary contacts and as a generative model. Before that, we take a closer look at another important strategy to identify protein sectors: Statistical Coupling Analysis.

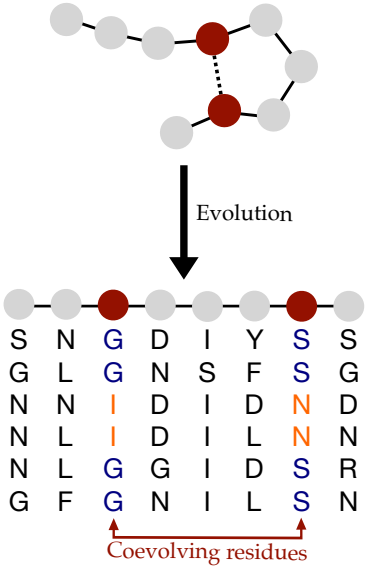

Figure 1.6: How evolutionary constraints influence sequence variability among homologs
At the top, a schematic illustrates a protein sequence folded in such a way that the two red residues are in physical contact. The evolutionary conservation of this contact induces coevolution between the two residues and gives rise to statistical correlations between the corresponding columns in the multiple sequence alignment.

7: Allostery refers to the thermodynamic coupling of distant sites without direct physical interaction and will be further discussed in Chapter 7.

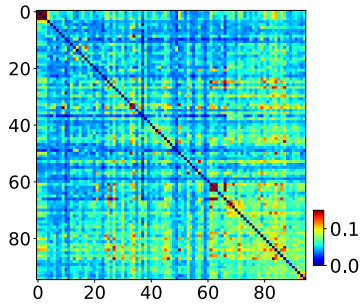

Figure 1.7: Correlation matrix of a Chorismate Mutase MSA.
Frobenius norm of the correlation matrix $C_{ij}(a,b)$ computed from an MSA of 1258 Chorismate Mutase sequences [30].

1.5.2 Protein sectors

Statistical Coupling Analysis (SCA) [19, 29, 39] is an approach originally inspired by methods applied in finance to study correlations in stock market data [40].

Given a multiple sequence alignment, the initial step of the procedure involves weighting the sequences to mitigate sampling biases. This important step is common to other approaches and will be discussed in more detail in Section 2.1.6.

SCA then uses the correlation matrix, defined in Equation 1.3, combined with a measure of amino acid conservation, to compute a matrix of

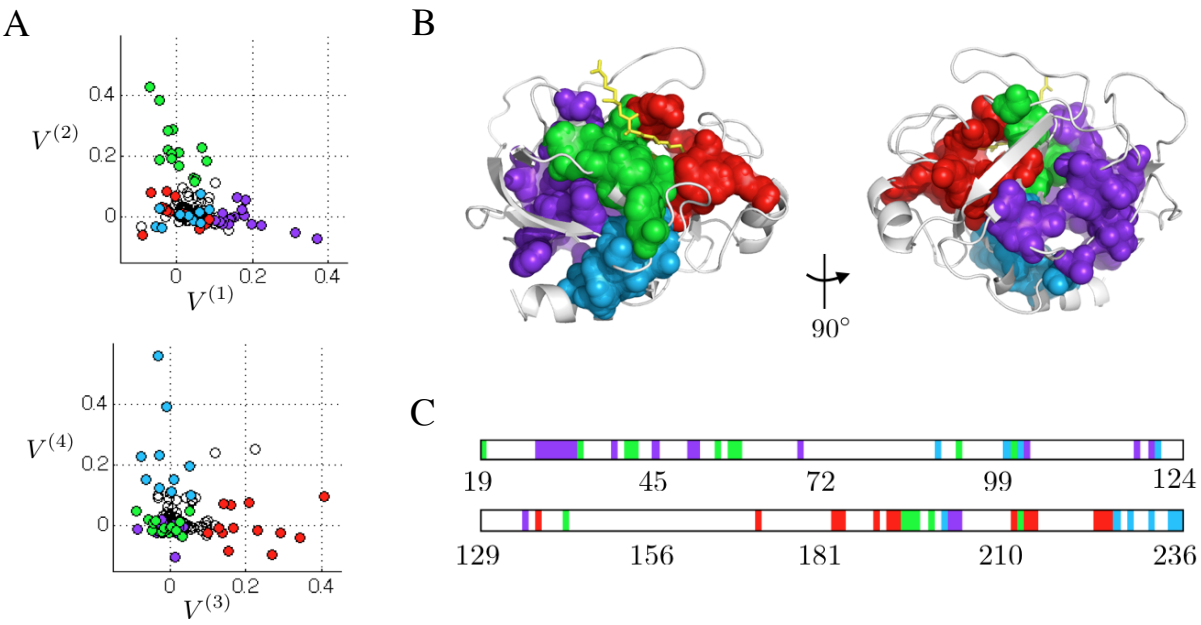

Figure 1.8: Protein sectors in the S1A family. (Figure from [32])
A. Projection of the sequence positions i along the vectors V^k, obtained by rotating the top four eigenvectors of the SCA matrix $\tilde{C}_{ij}$. Each sector k is defined as the set of positions whose projection along V^k exceeds a chosen threshold. Sectors are color-coded: purple ($k = 1$), green ($k = 2$), red ($k = 3$), and cyan ($k = 4$).
B. Two views of the Rat trypsin structure (PDB: 3TGI), with residues belonging to the four sectors highlighted in the same colors as in panel A. Notably, the sectors form spatially contiguous groups of residues that are in contact in the 3D structure.
C. Sequence positions of the residues belonging to each sector. As shown, sectors may span distant regions along the primary sequence.

conserved correlations, denoted $\tilde{C}_{ij}$. From this matrix, coevolving units are extracted through the following steps:

1. Computing the eigenvectors corresponding to the top eigenvalues of $\tilde{C}_{ij}$.
2. Applying Independent Component Analysis (ICA) to rotate these eigenvectors into statistically independent components.
3. Defining each sector as the set of positions contributing most strongly to a given component.

This methodology has since been applied to a broad range of protein families, and the resulting sectors have been tested experimentally in several systems. Notable examples include serine proteases [19], dihydrofolate reductase [36], and rhomboid proteases [41].

In Chapter 6 and Chapter 7, we focus on two such families, the serine proteases and the dihydrofolate reductases, in which SCA has identified sectors associated with distinct functional properties [19, 29, 32, 36].

As an example, Figure 1.8 presents results from a SCA performed by Olivier Rivoire on the serine protease family [32]. The contribution of each position to the different components is shown in Figure 1.8A, highlighting the independence of the resulting sectors. Analyzing the spatial and sequential distributions of these independent groups shows that they are in contact in the 3D structure, even though they are scattered along the sequence (Figure 1.8B & C).

More importantly, these sectors have been associated with specific biochemical properties. For example, the green sector has been experimentally linked to the catalytic activity of the enzyme, while the red sector is associated with substrate specificity, i.e., the enzyme's ability to selectively recognize and act on particular substrates.[8]

8: This property will be the main focus of Chapter 6.

1.6 Concluding remarks

In this chapter, we have outlined key features of proteins, focusing in particular on the concept of protein families—groups of sequences sharing similar structures and functions—and on how such sequences can be processed into multiple sequence alignments (MSAs) for statistical analysis.

We have also emphasized the importance of technological advances that have made it possible to collect sufficiently large datasets of protein sequences, enabling the application of statistical inference methods. One such method, which will serve as a central tool throughout this manuscript, is the Boltzmann Machine. It will be introduced and discussed in detail in the following chapter.

Generative models for protein sequences

2

The main goal of statistical inference is to estimate the N_p parameters $\boldsymbol{\theta} \in \mathbb{R}^{N_p}$ of a statistical model

$$P(D;\boldsymbol{\theta}) : \mathcal{D} \times \mathbb{R}^{N_p} \longrightarrow [0,1], \tag{2.1}$$

given a data set D^*, assuming that $D^* \sim P(D;\boldsymbol{\theta})$.

Statistical modeling therefore involves two main steps: (i) choosing a probabilistic model $P(D;\boldsymbol{\theta})$, and (ii) selecting an inference procedure to estimate $\boldsymbol{\theta}$ from the observed data D^*.

This thesis deals with statistical inference for protein sequences. The data consist of homologous proteins selected by natural evolution to perform a common biological function (see Section 1.4). In particular, we work with multiple-sequence alignments (MSAs) containing M sequences from the same family, each aligned to a common length L (Section 1.4.1).

Throughout this study, we adopt the Boltzmann Machine as our modeling framework. This chapter introduces this model and highlights the key challenges of training it and using it to design protein sequences.

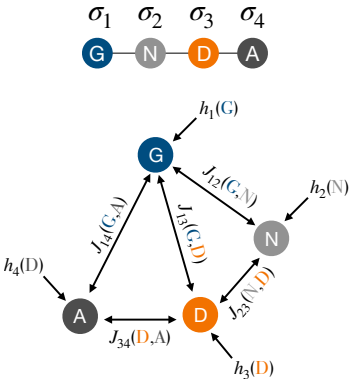

Figure 2.1: Schematic illustration of the Boltzmann-Machine architecture for a protein sequence.
The upper row shows a four-residue sequence, `GNDA`. The panel below depicts the corresponding undirected graphical model, where each vertex represents a residue. Pairwise couplings and the local fields acting on each vertex are indicated.

2.1 Boltzmann Machine

First proposed in 1983 by Ackley, Sejnowski, and Hinton[1], the Boltzmann Machine (BM) is an undirected graphical model, i.e. a probability distribution on a multidimensional space, defined by an interaction graph [42].

As shown in Figure 2.1, the graph contains a set of random variables $\boldsymbol{\sigma}$ that interact through a coupling matrix $\boldsymbol{J}$. In the context of protein sequences, each node corresponds to a sequence position and may assume $q = 21$ states (the 20 standard amino acids plus the gap).[2]

A configuration of the system is an amino acid sequence $\boldsymbol{\sigma} = \{\sigma_i\}_{i=1}^{L}$, whose probability follows the Boltzmann distribution

$$P(\{\sigma_i\}_{i=1,\dots,L}) = \frac{e^{-E(\sigma)}}{Z(\boldsymbol{h},\boldsymbol{J})}, \tag{2.2}$$

with statistical energy

$$E(\boldsymbol{\sigma}) = -\sum_{i=1}^{L} h_i(\sigma_i) \; - \sum_{i<j} J_{ij}(\sigma_i,\sigma_j), \tag{2.3}$$

and partition function

$$Z(\boldsymbol{h},\boldsymbol{J}) = \sum_{\tilde{\boldsymbol{\sigma}} \in \mathcal{S}} e^{-E(\tilde{\sigma})}. \tag{2.4}$$

The model parameters—the local fields $h_i(\sigma_i)$ and pairwise couplings $J_{ij}(\sigma_i,\sigma_j)$—must be inferred from data; the inference procedure is detailed in Section 2.1.2.

1: Hinton received the 2024 Nobel Prize in Physics, together with John Joseph Hopfield, for "foundational discoveries and inventions that enable machine learning with artificial neural networks."

2: Both the BM and the Restricted Boltzmann Machine (RBM) are special cases of the general architecture introduced in [42], which features a visible layer, a hidden layer, and full pairwise couplings among all variables.

Mathematically, the BM is equivalent to the Potts model, and estimating its parameters is known in statistical physics as the *Inverse Potts problem* [43]. Whereas conventional statistical physics predicts macroscopic observables from known microscopic laws, inverse problems proceed in the opposite direction: starting from empirical data, one seeks to reconstruct the underlying microscopic interactions.[3]

3: Inverse approaches appear in many fields, from neuroscience and finance to collective animal behaviour such as bird flocks [44, 45].

Notably, the Boltzmann-Machine model can also be derived from the maximum-entropy principle, which we introduce next.

2.1.1 A maximum entropy approach

Assume that the information contained in a data set can be summarised by a vector of statistics. Formulated in 1957 by Edwin Thompson Jaynes [46, 47], the maximum-entropy principle asserts that the least-biased model consistent with these statistics is the distribution that maximises Shannon entropy, subject to normalisation and to constraints on the expected values of the chosen statistics.

Given an MSA $\boldsymbol{\sigma} = \{\boldsymbol{\sigma}^{(m)}\}_{m=1}^{M}$ with M sequences, maximising the Shannon entropy

$$H(P) = -\sum_{\sigma} P(\sigma)\, \log P(\sigma) \tag{2.5}$$

under the normalisation condition $\sum_{\sigma} P(\sigma) = 1$ and under constraints on empirical single-site and pairwise amino acid frequencies leads to the Boltzmann distribution of Equation 2.2.

The $q \times L$ single-site constraints, for each position $i \in \{1, \dots, L\}$ and residue $a \in \{0, \dots, 20\}$, read

$$\sum_{\boldsymbol{\sigma}} P(\boldsymbol{\sigma})\, \delta(\sigma_i, a) = \frac{1}{M} \sum_{m=1}^{M} \delta(\sigma_i^{(m)}, a) = f_i(a), \tag{2.6}$$

while the $q^2\, L(L-1)/2$ pairwise constraints[4] are

$$\sum_{\boldsymbol{\sigma}} P(\boldsymbol{\sigma})\, \delta(\sigma_i, a)\, \delta(\sigma_j, b) = \frac{1}{M} \sum_{m=1}^{M} \delta(\sigma_i^{(m)}, a)\, \delta(\sigma_j^{(m)}, b) = f_{ij}(a, b). \tag{2.7}$$

4: Because $f_{ij}(a, b) = f_{ji}(b, a)$.

Introducing Lagrange multipliers for each constraint yields the distribution in Equation 2.2; the fields and couplings act as the multipliers and are chosen so that Equation 2.6 and Equation 2.7 are satisfied [48].[5]

5: Viewed through the lens of inference, the Gibbs-Boltzmann distribution follows directly from the maximum-entropy principle. Statistical mechanics, however, is a physical theory; borrowing its formalism does not imply the same physical interpretation.

Although the model in Equation 2.2 results from the specific constraints introduced above, alternative models can be derived by enforcing different sets of constraints. The question of how to choose relevant observables within the maximum entropy framework is addressed later in Section 2.1.7.

2.1.2 Maximum Likelihood Estimation (MLE)

The inference of the Boltzmann Machine model introduced in Section 2.1 can be stated as:

Definition 2.1.1 ***(BM inference from an MSA)***
Given the empirical single-site frequencies $f_i(a)$ and pairwise frequencies $f_{ij}(a, b)$ computed from a multiple sequence alignment, BM inference consists in finding the parameters $\boldsymbol{\theta}^* = \{\boldsymbol{J}^*, \boldsymbol{h}^*\}$ such that the constraints in Equation 2.6 and Equation 2.7 are satisfied.

This inference problem naturally fits within the framework of Maximum Likelihood Estimation (MLE). Given a multiple sequence alignment $\{\boldsymbol{\sigma}^{(m)}\}_{m=1}^{M}$ and a model $P(\boldsymbol{\sigma} \mid \boldsymbol{\theta})$, the maximum likelihood principle prescribes choosing the parameters that maximise the average log-likelihood[6]:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{M} \sum_{m=1}^{M} \log P(\boldsymbol{\sigma}^{(m)} \mid \boldsymbol{\theta}), \tag{2.8}$$

where $\mathscr{L}(\boldsymbol{\theta}) = \frac{1}{M} \sum_{m=1}^{M} \log P(\boldsymbol{\sigma}^{(m)} \mid \boldsymbol{\theta})$ denotes the average log-likelihood of the data.

It can be shown that the inference problem in Definition 2.1.1 follows directly from this principle. The $q \times L + q^2 \frac{L(L-1)}{2}$ constraints in Equation 2.6 and Equation 2.7 are recovered by setting the derivatives of $\mathscr{L}(\boldsymbol{\theta})$ with respect to the model parameters to zero:

$$0 = \frac{\partial \mathscr{L}(\boldsymbol{\theta})}{\partial h_i(a)} = \sum_{\boldsymbol{\sigma}} P(\boldsymbol{\sigma}) \delta(\sigma_i, a) - f_i(a), \tag{2.9}$$

$$0 = \frac{\partial \mathscr{L}(\boldsymbol{\theta})}{\partial J_{ij}(a, b)} = \sum_{\boldsymbol{\sigma}} P(\boldsymbol{\sigma}) \delta(\sigma_i, a) \delta(\sigma_j, b) - f_{ij}(a, b). \tag{2.10}$$

However, a major difficulty arises at this point: evaluating the likelihood requires computing the partition function $Z(\boldsymbol{\theta})$, which involves summing over all possible sequences:

$$Z(\boldsymbol{\theta}) = \sum_{\tilde{\boldsymbol{\sigma}} \in \mathcal{S}} e^{-E(\tilde{\boldsymbol{\sigma}})}.$$

This sum contains q^L terms, which is intractable in practice for typical protein lengths[7].

As a result, numerical approximations are required. In the following section, we describe in more detail the optimisation procedure used throughout this thesis.

6: This is equivalent to minimising the Kullback–Leibler divergence between the empirical distribution and the model distribution.

7: One might say that computing the partition function is *the* problem of statistical mechanics.

2.1.3 Gradient descent optimization

Assuming that the objective is to minimise the negative *log-likelihood*, the model parameters can be updated using gradient descent:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \eta_t \boldsymbol{p}_t, \tag{2.11}$$

where $\boldsymbol{p}_t$ denotes the search direction, η_t the learning rate, and $\boldsymbol{\theta}_t$ is the parameter vector at iteration t (of dimension N_p).

The choice of $\boldsymbol{p}_t$ depends on the optimisation algorithm. In the case of Vanilla Gradient Descent (VGD), it corresponds to the gradient of the objective function, as given by Equation 2.9 and Equation 2.10.

The inference procedure involves generating N_{chains} synthetic samples from the model at each iteration t, using Markov Chain Monte Carlo (MCMC) methods such as Gibbs sampling [49] or Metropolis–Hastings [50, 51].[8] These samples are used to estimate the model expectations and thus approximate the gradient needed to update the parameters.

8: A short description of the Metropolis–Hastings algorithm used throughout this thesis is provided in Section A.2

A pseudocode version of the algorithm is provided below. The main hyperparameters are summarised in the schematic representation shown in Figure 2.2.

Algorithm 1: Boltzmann Machine learning (pseudocode)

Input: $\boldsymbol{\theta}_0$, N_{chains}, N_{iter}[9]

for $t = 1, \ldots, N_{\text{iter}}$: **do**
  Generate N_{chains} sequences $\sim P(\boldsymbol{\sigma} \mid \boldsymbol{\theta}_t)$ using MCMC;
  Compute $\boldsymbol{p}_t$ from the N_{chains} samples;
  $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \eta_t \boldsymbol{p}_t$;
end

9: Alternatively, training can be stopped once a convergence criterion is met, such as a sufficiently small gradient norm. Note that the log-likelihood cannot be monitored directly during training.

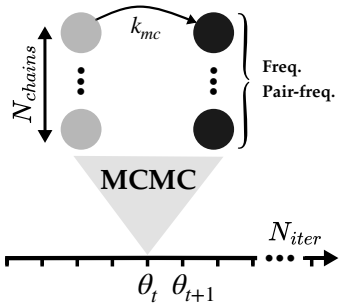

Figure 2.2: Schematic illustration of the gradient descent algorithm.
At each iteration t, model parameters are stored in the vector $\boldsymbol{\theta}_t$. A Monte Carlo simulation is then used to generate N_{chains} samples from the current model. After k_{mc} MCMC steps, the sampled sequences are used to compute single-site and pairwise frequencies, which are needed to estimate the gradient of the objective function. The N_{chains} Markov chains are run in parallel, each for the same number of steps.

The algorithm starts from an initial guess for the parameters. While this choice is arbitrary in principle, a good initialisation can considerably accelerate convergence. A common strategy is to initialise with the parameters of an independent model, where pairwise couplings are set to zero.

Once the model has been trained, it can be used to sample artificial sequences (see Section 2.1.10), for which single-site and pairwise frequencies are expected to match those of the training data, as they are explicitly fitted. Interestingly, it has been shown that the model also captures additional statistics that are not imposed during training. For instance, third-order correlations are often well reproduced, and Principal Component Analysis (PCA) yields similar distributions for artificial and natural sequences. The apparent statistical indistinguishability between model-generated and natural sequences is often taken as evidence that the Boltzmann Machine serves as an effective generative model for protein sequences. We postpone to Section 2.3 a thorough discussion of what constitutes a good generative model.

Nevertheless, several computational and statistical challenges remain, which are the subject of ongoing research. These issues will be discussed in the following sections.

2.1.4 The MCMC bottleneck

In practice, the efficiency of the learning procedure, as well as the quality of the inferred model, is often limited by the MCMC sampling step, which becomes increasingly demanding as sequence length grows.

In theory, proper sampling requires that the number of MCMC steps exceeds the mixing time of the system. However, studies on standard datasets such as MNIST have shown that the mixing time is typically large and tends to increase over the course of training [52]. As a result,

even with large values of k_{mc}, training is likely to operate under non-equilibrium conditions, which can affect the properties of the learned model.

Interestingly, Decelle *et al.* [52] have shown that this out-of-equilibrium regime can be leveraged to efficiently generate high-quality samples: by initializing MCMC chains from random sequences, using a small k_{mc}, and generate synthetic data using the same number of MCMC steps as employed during training.[10] On the other hand, if the goal is to match the equilibrium distribution of the model to that of the dataset, k_{mc} must be set much larger than the mixing time, which is computationally expensive.

10: When this last condition is not met, i.e., when sampling is performed at equilibrium rather than under the same conditions as training, a reduction in sample diversity is typically observed. While this observation was made for RBM models, it likely extends to MCMC-MLE approaches more generally. This aspect is further explored in Section A.3, in the context of a BM trained on protein sequences.

Besides resorting to inference methods that bypass MCMC sampling (see Section 2.1.9), several strategies have been proposed to reduce the computational cost of training these models.

One such method is Contrastive Divergence (CD), which consists in initializing the MCMC chains with natural sequences from the dataset. This is based on the assumption that natural sequences are good approximations of equilibrium samples for a well-trained model [53].

Another widely used approach, called Persistent Contrastive Divergence (PCD), reuses the final states of the chains from the previous iteration [54]. This method assumes that if the parameters change smoothly during training, the equilibrium distribution evolves accordingly, and only a few MCMC steps are needed to maintain approximate equilibrium. However, PCD tends to explore the distribution locally and may remain trapped in a single mode, especially when the probability landscape becomes sharply peaked.

Regardless of the exact MCMC strategy used, the training of BM models (and MCMC-MLE methods in general) is complex and remains an active topic of research.

2.1.5 The undersampling bottleneck

Inverse Potts models are typically inferred from MSAs containing $M \sim 10^2$–10^5 sequences, which is much smaller than the size of the parameter space, scaling as $q^2L^2 \sim 10^5$–10^7. As a result, any inference method, even those discussed later in Section 2.1.10, must be combined with regularization techniques to prevent overfitting or to make the problem tractable in practice.

This crucial aspect of model inference is the central focus of the second part of this thesis. In particular, we will briefly review the most common regularization strategies in Section 3.2, discuss the undersampling-induced biases reported by Kleeorin *et al.* [30] in Section 3.3, and introduce a solution to mitigate these effects in Chapter 4.

2.1.6 Phylogenetic biases

A key assumption underlying BM modelling is that homologous protein sequences are independently sampled from an equilibrium distribution. In practice, this assumption is clearly violated due to the shared evolutionary history of the sequences, which introduces correlations that arise not from functional or structural constraints, but from phylogenetic relationships [55].

A common strategy to mitigate this bias is to compute weighted statistics [56]:

$$f_i(a) = \frac{1}{M_{\text{eff}}} \sum_{m=1}^{M} w^{(m)} \delta(\sigma_i^{(m)}, a), \tag{2.12}$$

$$f_{ij}(a,b) = \frac{1}{M_{\text{eff}}} \sum_{m=1}^{M} w^{(m)} \delta(\sigma_i^{(m)}, a) \delta(\sigma_j^{(m)}, b), \tag{2.13}$$

where the weight $w^{(m)}$ is inversely proportional to the number of sequences that share more than α% identity with sequence m. The quantity $M_{\text{eff}} = \sum_{m=1}^{M} w^{(m)}$ is referred to as the effective number of sequences, accounting for redundancy in the dataset. This effective count depends on the identity threshold α, which is typically set around 70–80%.

2.1.7 The model specification problem

As mentioned in Section 2.1.1, alternative models can be derived by enforcing different sets of constraints within the maximum entropy framework. This raises a fundamental question: what is the most appropriate choice of statistical constraints?

Imposing only the single-site frequencies leads to an independent, or *profile*, model. These models, often combined with tools such as Hidden Markov Models, are widely used to align sequences to a given family. However, independent models have been shown to fail at generating functional sequences in experimental assays [57, 58].

Interestingly, constraining only single-site and pairwise frequencies already allows the BM model to reproduce higher-order statistics when trained on protein sequences. However, this does not imply that all one- and two-site statistics are important, or that collective variables could not provide a more relevant description. Several studies have therefore introduced methods to remove couplings considered as spurious [59, 60]. These so-called decimation, or pruning, schemes may be viewed as a practical form of regularisation.

2.1.8 Gauge invariance

Because frequencies sum to one, the number of independent constraints is smaller than the number of model parameters. As a result, the probability distribution remains unchanged under certain transformations known as *gauge* transformations.

Several conventions exist to fix the gauge of the model. In this manuscript, we will adopt the commonly used *zero-sum gauge* [34], also referred to as the *Ising gauge*, which imposes[11]:

$$\sum_a h_i(a) = \sum_a J_{ij}(a,b) = \sum_b J_{ij}(a,b) = 0. \tag{2.14}$$

11: This is the gauge that we will use to compare couplings of models trained with different inference procedures. For further technical details on this gauge transformation, see Section A.1

This choice of gauge has the effect of minimising the Frobenius norm of the coupling matrix, defined over amino acid indices as:

$$\|J_{ij}\| = \sqrt{\sum_{a,b} J_{ij}(a,b)^2}. \tag{2.15}$$

2.1.9 Other inference procedures

The BM learning approach [58, 61, 62] we have just described is neither the first nor the only method that has been used to estimate the parameters of the inverse Potts model in the context of protein sequences. In the following, we briefly review several important inference procedures that have been applied in this field.

Mean-field (MF) approximation [38]:
This was one of the first methods applied to inverse Potts model inference and remains among the most computationally efficient. In general terms, if the system consists of interacting units, the mean-field approximation assumes that the joint probability distribution can be factorized over these units. A generalisation of the standard MF approach, known as the Plefka expansion [63] or small-coupling expansion, leads to self-consistent equations in which estimating the coupling matrix reduces to inverting the empirical correlation matrix (Equation 1.3).[12]
This method has been successfully applied to the prediction of contacts in protein 3D structures. However, under this approximation, the statistical constraints given in Equation 2.6 and Equation 2.7 are not satisfied, and the model does not reproduce the empirical statistics.

12: Another mean-field approach, introduced in 2013 [31], reduces the number of parameters by introducing a low-rank representation of the coupling matrix.

Gaussian approximation [64]:
In this approach, protein sequences are represented as real-valued vectors, and a multivariate Gaussian distribution is defined over this continuous space. The mean and covariance matrix of the distribution are inferred within a Bayesian framework. As in the mean-field approximation, the coupling matrix is ultimately obtained by inverting the estimated covariance matrix.

Pseudo-likelihood maximization [34, 65]:
This inference method approximates the global *likelihood* function by a product of single-site conditional probabilities, defining the so-called *pseudo-likelihood* function. This approach is computationally efficient since the gradients can be computed exactly and has been shown to outperform the mean-field approximation in contact prediction tasks. Moreover, it can be shown that for Gibbs distributions, the estimators obtained from pseudo-likelihood converge to the maximum likelihood estimators in the infinite sampling limit.[13]

13: However, maximizing each local likelihood independently yields asymmetric couplings, i.e., $J_{ij} \neq J_{ji}$. In practice, the average of the two is used for contact prediction (see Section 2.1.10)

Autoregressive model [66]:
This approach relies on factorizing the joint probability of a sequence into a product of conditional probabilities using Bayes' rule. Inference can then be carried out similarly to pseudo-likelihood maximization, with exact computation of gradients. According to [66], this method offers comparable performance to standard BM inference, while being significantly more efficient computationally.

2.1.10 Applications to protein sequences

While the probability distribution of the BM model offers a natural framework for generating synthetic sequences, it has also been widely applied to the prediction of inter- and intra-protein contacts, as well as the evaluation of mutational effects. In what follows, we briefly review the main applications of this model, also commonly referred to as Direct Coupling Analysis (DCA) in the literature.

Contact prediction [38, 56, 67, 68]:
This is perhaps the most well-known application of the DCA method,

as it marked one of the first major advances toward predicting protein structures directly from sequence data. It has also proven useful for identifying physical contacts between interacting proteins.[14]
The standard strategy involves ranking residue pairs according to the Frobenius norm of the coupling matrix (Equation 2.15) and considering the top-scoring pairs as contacts in the 3D structure. To further improve predictive accuracy, a correction known as the Average Product Correction (APC) is often applied to remove background noise in the coupling parameters [71].[15]
This methodology has been validated across a wide range of protein families and has proven robust to both the choice of family and the inference method used. Beyond individual proteins, DCA has also been extended to identify interacting paralogs [72, 73].[16]

14: Note that early attempts to predict residue-residue contacts from coevolutionary information date back to 1994, using simple covariance measures [69]. One year later, mutual information was also employed for contact prediction [70]. These early approaches were eventually outperformed by DCA in 2009, a progress also made possible by the larger sequence datasets available at that time.

15: This correction has been claimed to reduce phylogenetic bias, but has recently been shown to also improve the identification of isolated contacts in a toy model without phylogeny in an undersampling regime [30].

16: Paralogs are homologous genes that result from duplication events within a genome. In this context, the common approach is to define a joint probability over concatenated sequences that may or may not interact.

Mutational effects prediction [66, 74–77]:
The statistical energy provided by the BM model can be used not only to score entire sequences, but also to quantify the effect of individual mutations. Typically, predictions focus on single-site mutations introduced into a natural homolog for which experimental data are available, often obtained through Deep Mutational Scanning (see Section 5.2.1).
In Chapter 5, we introduce key concepts related to protein properties and mutational effects, and in Chapter 6 and Chapter 7, we investigate the ability of the BM model to predict compensatory mutations in protein sequences.

Protein design:
An appealing application of the BM model is its ability to generate novel protein sequences by sampling from its probability distribution using MCMC simulations.
However, this was not the first attempt at artificial protein design. In 2005, Statistical Coupling Analysis (SCA) (see Section 1.5.2) was used to generate artificial sequences of the WW domain, which is 35 residues long [57, 78]. These studies highlighted the importance of capturing correlation signals to generate functional sequences.
More recently, the BM model (referred to as bmDCA) was shown to successfully generate functional sequences for the Chorismate Mutase family ($L \sim 100$) [58].[17] However, the sequences were not sampled directly from the original BM model but from a rescaled version, which fails to reproduce the full diversity observed in natural datasets (see Section 3.4.2). This limitation is related to the undersampling issue and will be the main focus of Chapter 3 and Chapter 4.

17: In the same study, the independent model, which reproduces only single-site frequencies, was unable to generate functional sequences.

2.2 Landscape of generative models for protein sequences

Over the past three decades, the literature on generative models has expanded dramatically. Many of the deep-learning architectures now applied to biological data originate in image and language generation. Their recent success for modelling protein sequences has been made possible by advances in high-throughput sequencing, which have enable to produced databases containing billions of protein sequences.

Below, we sketch the main actors in this broad landscape; for a comprehensive overview, the reader is referred to the following review [79].

Variational auto-encoders (VAEs) have been adapted to protein sequences in several studies [80, 81]. VAE couples a probabilistic encoder, which

maps each input into a distribution in a low-dimensional latent space, with a decoder that reconstructs data by sampling from this latent representation [82]. Hawkins-Hooker *et al.* trained such an architecture on luciferase sequences ($L = 360,\ M = 70{,}000$)—enzymes responsible for bacterial bioluminescence—and generated novel variants, up to 35 residues distant from natural homologues, whose luminescence was experimentally confirmed [80].

Generative Adversarial Networks (GANs) confront two neural networks in a minimax game: a generator attempts to mimic the training distribution while a discriminator learns to tell real from synthetic samples [83]. Repecka *et al.* applied such an architecture to malate dehydrogenase sequences ($L = 328,\ M = 16{,}000$); of 55 GAN-designed variants tested in *E. coli*, 24% retained measurable catalytic activity [84].

No less influential are *Transformer-based models*, whose impact on AI research has been considerable [85]. Initially developed for natural-language processing, a Transformer processes input sequences in parallel through stacked encoder–decoder blocks built around the self-attention mechanism, allowing each token to attend to every other token and thereby capture long-range dependencies [86]. Transformers have since been repurposed as generative models for proteins [87–89]; they are typically pre-trained on massive protein databases before being fine-tuned on a specific family.

Finally, *diffusion models*, which add noise through a forward process and train a neural network to reverse it step by step [90], have also entered the protein-design arena. Watson *et al.*, for example, combined a diffusion framework with Rosetta refinements to generate proteins directly in three-dimensional coordinate space, achieving successful *de novo* designs [91].

All these neural architectures learn the data distribution implicitly, optimising parameters with no explicit link to the statistics they reproduce. By contrast, the Boltzmann Machine framework adopted in this thesis is specified *explicitly*: there is a set of observables (here, single-site and pairwise amino acid frequencies) and the model is constrained to match their empirical values. This construction, rooted in the maximum-entropy principle, yields a transparent energy function whose parameters are interpretable.

2.3 Assessing Generative Model Quality

There is an elephant in the room when it comes to unsuppervised generative modeling: evaluating model performance remains a fundamentally difficult problem. Unlike supervised learning, there is no obvious metric, such as classification accuracy on a test set, that can be computed to assess how well the model performs.[18]

18: Although some supervised evaluation techniques can be adapted to generative contexts, they generally do not provide a complete picture of generative performance.

Broadly speaking, the evaluation criteria for generative models might be grouped into the following categories [92]:

- **Fidelity**: Do the generated samples exhibit high quality, i.e., do they share key properties with those observed in the training data? In the case of proteins, this typically includes the ability to fold into stable structures and to perform biological functions.

- **Novelty**: Are the generated sequences meaningfully different from those in the training data? This criterion reflects the model's capacity to generalize beyond what it has seen.
- **Diversity**: Does the model capture the full range of variation present in the training set?[19]

19: For instance, a model that only replicates training sequences may produce high-quality samples and preserve diversity, but lacks novelty.

These criteria are broadly applicable across domains. However, the methods used to evaluate them are sometimes domain-specific and cannot always be directly transferred, particularly when it comes to assessing fidelity.

In the following, we review some of the common strategies used to assess generative models for protein sequences. Before doing so, we first highlight the importance of data partitioning as a key tool for model evaluation.

2.3.1 Data partitioning

Data partitioning is a preliminary step in the evaluation of machine learning models. A common strategy is to divide the dataset into a training set, used to learn model parameters, and a test set, used to assess the model's ability to generalize [93].[20]

20: A validation set may also be used for hyperparameter tuning

In the case of generative models for protein sequences, procedures for constructing training and test sets are not always clearly described in the literature. Since the goal is to evaluate generalization, it seems reasonable to ensure that the test set does not contain sequences identical to those in the training set. For BM models, where sequence weighting is used to mitigate phylogenetic biases (Section 2.1.6), a natural choice would be to rely on the same identity threshold used for the weighting procedure to define whether two sequences should be considered identical from the model's perspective.

A practical way to construct the training and test sets is, for example, to cluster sequences that exceed this identity threshold and then randomly assign these clusters to either the training or test set.

2.3.2 Fidelity

In the context of protein sequences, fidelity addresses the question: *do the generated sequences exhibit the characteristics expected of natural proteins?*

Ultimately, this relates to the ability of a sequence to fold and carry out a specific biological function. Wet-lab characterisation is thus considered as the gold standard for generative model validation. However, such experiments are costly, require technical expertise and inherently context dependent[21]. There are many experimental measurements one could make, several of which depend on the specific function of the protein under study, and it is not always obvious which of them provides the most meaningful assessment. Consequently, while some studies do include experimental validation [57, 58, 80, 81, 91, 94, 95], most rely primarly on in-silico analyses.

21: Natural homologues themselves rarely function under identical conditions

A natural starting point is to compare the statistical properties of synthetic sequences with those of natural ones. While the BM model is explicitly fitted to reproduce single-site and pairwise frequencies, one can further evaluate its performance by examining *unconstrained* features, such as third-order correlations. Another common approach involves performing

principal component analysis (PCA) or t-SNE method on the natural sequences and projecting both natural and generated sequences into the resulting low-dimensional space, to assess whether the distributions are similar [43, 66, 80, 88, 96, 97]. Importantly, such comparisons should be made with respect to the discrepancy observed between the training and test sets: the model should match the test-set statistics as well as the training set does.

A complementary approach leverages the model's statistical energy or more generally, its capacity to assign scores to sequences [58, 66, 88]. Random sequences should consistently receive higher energies (i.e., lower probabilities) than either natural sequences or those generated by the model. Similarly, sequences drawn from a profile model can serve as a negative control: if they are assigned scores comparable to natural or synthetic sequences, this suggests that the BM model is not leveraging pairwise couplings to improve over the independent baseline.

Another frequent strategy is to evaluate generated sequences using external predictive models [80, 88, 89, 98]. For instance, structure prediction models like AlphaFold2 [14] or Rosetta [99] produce confidence scores and global similarity metrics that reflect how likely a sequence is to fold properly. Likewise, profile-based models such as HMMER [27] can be used to evaluate whether synthetic sequences would plausibly be classified within the target protein family.

2.3.3 Novelty

The notion of novelty relates to the model's capacity to generalize, i.e., to produce sequences that differ from those seen during training.

Measuring novelty requires a way to quantify sequence similarity, for which the Hamming distance is often used:

$$d_H(\boldsymbol{\sigma}, \boldsymbol{\tau}) \;=\; \frac{1}{L}\sum_{i=1}^{L}\big(1-\delta(\sigma_i,\tau_i)\big), \tag{2.16}$$

where $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$ are two sequences of length L and δ is the kronecker symbol. Throughout this manuscript, we will rather report the identity percentage, defined as $100 \times (1 - d_H)$.

A simple way to assess novelty is to calculate this identity between each synthetic sequence and its closest natural counterpart. For natural sequences, the same procedure is applied while excluding self-comparisons. If sequence weighting was used to mitigate phylogenetic bias in training (see Section 2.1.6), it may also be appropriate to weight the distribution of identities for natural sequences accordingly.

In addition to sequence similarity, the model's scoring capacity offers another test of generalization. Specifically, one can verify that test sequences receive scores comparable to those assigned to training sequences. This idea is analogous to verifying that a supervised classifier generalizes well to previously unseen data.

Finally, another intersting metric is provided by the heat capacity of the model, defined as:

$$C = \frac{\partial \langle E \rangle_T}{\partial T} = \frac{\langle E^2 \rangle_T - \langle E \rangle_T^2}{K_B T^2}, \tag{2.17}$$

where is T is the temperature and the averages $\langle . \rangle_T$ are evaluated by sampling with the Boltzmann weight $\exp(-E/T)$. This quantity is typically plotted against temperature and large values of heat capacity indicate large variations of the model entropy with T. In unregularized models, it has been observed that the heat capacity typically peaks near $T = 1$, the training scale, which is a signature of criticality[22] and suggests that the model is overfitting. In contrast, regularized models tend to exhibit less pronounced peaks, indicating more robust learning [59, 100].

22: Note, however, that we cannot determine whether the observed peaks correspond to a genuine phase transition in the thermodynamic limit.

2.3.4 Diversity

Evaluating diversity involves assessing whether the generated sequences span the same range of variability as the training data.

A first approach is to compute the pairwise identity distributions within both the natural and artificial sequence sets and compare them. Significant deviations may indicate either mode collapse or overdispersion.

PCA-based visualizations can offer a complementary view. By projecting both natural and synthetic sequences onto the first two principal components extracted from the natural data, one can observe whether the artificial ensemble populates the same sequence subspace. However, caution is needed when interpreting these plots, as the first two components typically explain only a small fraction of the total variance in practice.

2.3.5 Employing a Toy Model

Employing a toy model, or synthetic model is an interesting way to evaluate generative models. In this case, a model for which we know the parameters is used to generate sequences that will be used as the training sequences for another model. This setup has the advantage of providing access to the "ground truth" model, allowing for direct comparisons between inferred and true parameters.

In many studies, models are inferred from synthetic sequences generated by a previously trained model (typically another Boltzmann Machine) that was itself trained on real protein sequence data [88, 101–103]. Yet this method assumes that the generative model used to produce synthetic data has accurately captured the structure of the original dataset without introducing significant biases. The issue of biases arising from the undersampling regime will be the central focus of Chapter 3 and Chapter 4.

Rather than relying on synthetic data generated from inferred models, one may instead construct a toy model explicitly designed to encode features believed to be crucial for protein sequences. While this approach offers greater interpretability and control, it comes with the inherent risk of incorrectly identifying which features are essential or not.

Kleeorin *et al.* [30], for example, proposed a toy model that incorporates a network of multiscale interactions, which in real data play a fundamental role in structure, function, and evolvability. This design is motivated by extensive evidence that, beyond local residue-residue contacts, co-evolving groups of residues are also crucial for protein function (see Section 1.5.1). Some of these groups, commonly referred to as sectors (see Section 1.5.2), have been experimentally characterized in a range of systems, including serine proteases [19], dihydrofolate reductases [36], and rhomboid proteases [41]. This toy model introduced by Kleeorin *et al.*

will serve as a central framework for the analyses carried out in Chapter 3 and Chapter 4 of this thesis.

2.4 Concluding remarks

This chapter was devoted to the main character of this thesis: the Boltzmann Machine model.

In particular, we examined the key challenges of training this model and using it as a generative tool for protein sequences. We chose to conclude this chapter with a discussion on the evaluation of generative models for protein sequences. The overall picture that emerges highlights the need to approach this evaluation as a multi-dimensional problem.

Throughout this manuscript, the expression *generative capacity* will thus always refer to this broader perspective and not simply, for example, to a model's ability to generate a significant number of functional sequences.

Finally, stating that BM models are efficient generative models can obscure the critical role played by the inference procedure, especially the regularization strategy, in shaping the resulting model. This point will be the focus of Chapter 3 and Chapter 4, and whenever we refer to the generative capacity of the BM model, it should be understood in the context of a specific inference setting.

The Undersampling Problem

Undersampling-induced biases | 3

The second part of this thesis is dedicated to the undersampling problem in generative models for protein sequences, with a particular focus on the Boltzmann Machine introduced earlier.

Chapter 3 introduces the undersampling-induced biases highlighted in a paper published one year before the start of my PhD by Kleeorin *et al.* [30]. It should be noted that the results presented in this chapter do not come directly from the original paper but have been reproduced using the model employed throughout this manuscript to ensure a consistent framework.

Then, Chapter 4 details a new solution to correct the undersampling-induced biases: the Stochastic Boltzmann Machine (SBM).

3.1 Undersampling regime in sequence data

The aim is to model the sequences of a protein family as samples drawn from a probability distribution. In the case of the Boltzmann Machine (BM), this distribution is given by:

$$P(\{\sigma_i\}_{i=1,\dots,L}) = \frac{1}{Z(\boldsymbol{h},\boldsymbol{J})} \prod_{i=1}^{L} e^{h_i(\sigma_i)} \prod_{i<j} e^{J_{ij}(\sigma_i,\sigma_j)}, \tag{3.1}$$

where the parameters h_i and J_{ij} are each associated with a statistical constraint of the model.

As previously noted, the protein sequences collected into MSAs typically span lengths of $L \sim 50$ to 500, and the alphabet size is fixed at $q = 21$, corresponding the 20 amino acids plus the gap character. As a result, the number of parameters in the distribution defined by Equation 3.1 grows approximately as $q^2L^2 \sim 10^5 - 10^7$.

A maximum entropy model without any constraints assigns equal probability to the $q^L \sim 10^{66} - 10^{650}$ possible sequences. Once statistical constraints are introduced, the model will favor sequences that better reflect the empirical statistics derived from the MSA, assigning them higher probabilities, while disfavoring those that deviate.

However, these empirical statistics are typically estimated from MSAs containing only a few hundred sequences, or up to $M \sim 10^5$ in the most favorable cases. Ideally, to capture all possible amino acid combinations at least once—assuming no redundancy—the number of sequences would need to exceed the size of the parameter space. For example, as illustrated in Figure 3.1, over 68% of all possible amino acid combinations are missing from an alignment of 1258 Chorismate Mutase sequences of length 96.

The situation is further complicated by the evolutionary relatedness among sequences and the over-representation of certain species in sequencing databases, both of which introduce significant sampling biases. To mitigate these biases, a standard approach is to weight the importance of each sequence in the calculation of the empirical statistics (see Section 2.1.6).

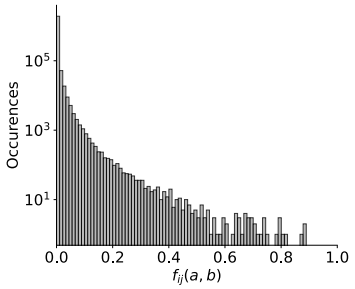

Figure 3.1: Pairwise frequencies for a Chorismate Mutase alignment. Histogram of pairwise frequencies computed from a MSA of the Chorismate Mutase family, made by Kleeorin *et al.* [30]. In this alignment, which includes 1258 sequences and 96 positions, we find that more than 10^6 pairwise combinations—representing 68% of all possible pairs—are not observed.

Such observations have very tangible effects for the inference of Boltzmann Machine (BM) models. From the model's perspective, amino acid combinations not observed in the alignment correspond to infinitely negative couplings. As a result, any sequence containing such combinations is assigned a near-zero probability. Since many of these absences arise from the undersampling regime, the model's capacity to generalize beyond the observed data is severely limited.

Figure 3.2A shows the results of a model trained on only a subset of a chorismate mutase family alignment. Many of the "test" sequences, which were excluded from the empirical statistics on which the inference is based, have much higher statistical energies, i.e. lower probabilities with respect to the model, than the training sequences. Notably, statistical energy of test sequences correlates inversely with identity to the nearest natural sequence: the less similar a sequence is, the higher the energy assigned by the model.

This discrepancy between training and test datasets is a hallmark of overfitting, which not only affects model performance on unseen data but also compromises robustness: models trained on different datasets may have significantly different parameter estimates. This common challenge in machine learning is typically mitigated through regularization techniques.

3.2 Regularization

Various regularization methods have been proposed to produce more robust and generalizable models [93]. Below is a non-exhaustive list of common approaches used in Boltzmann Machines:

- **Pseudo-counts**: This method helps avoid issues with boundary statistics. For example, observed frequencies can be adjusted as $f_i(a) \to (1-\alpha)f_i(a) + \frac{\alpha}{q}$. In practice, this means augmenting the MSA with a fraction $\alpha/(1-\alpha)$ of sequences where amino acids are drawn uniformly at random [38, 75].[1]
- **Decimation or pruning**: These procedures iteratively remove parameters considered irrelevant, often focusing on couplings, either during or after training [59, 60, 104]. The decimation can also be done before the inference by considering only statistics that exceed a certain threshold.
- **Alphabet reduction**: Also called "color compression", these approaches consist in grouping rare amino acids into a single state at particular sites [97, 105]. Other reduction schemes are based on physiochemical properties of the amino-acids [106].
- **L_p-norm regularization**: These approaches penalize the log-likelihood with a term proportional to the L_p-norm of the fields and couplings. The L_1-norm promotes sparse solutions, while the L_2-norm discourages large parameter values but does not enforce sparsity, allowing all parameters to contribute to the model [33, 34, 93].

1: A pseudo-count $\alpha = 1$ corresponds to a uniform model in which all frequencies are equal to $1/q$ and pairwise frequencies are equal to $1/q^2$.

These regularization techniques can be applied differently to first-order and second-order statistics, which correspond to fields and couplings, respectively. Since the second-order statistics are more sensitive to undersampling, it is common practice, for instance, to use stronger regularization on the couplings or to apply the pruning procedures exclusively to them.

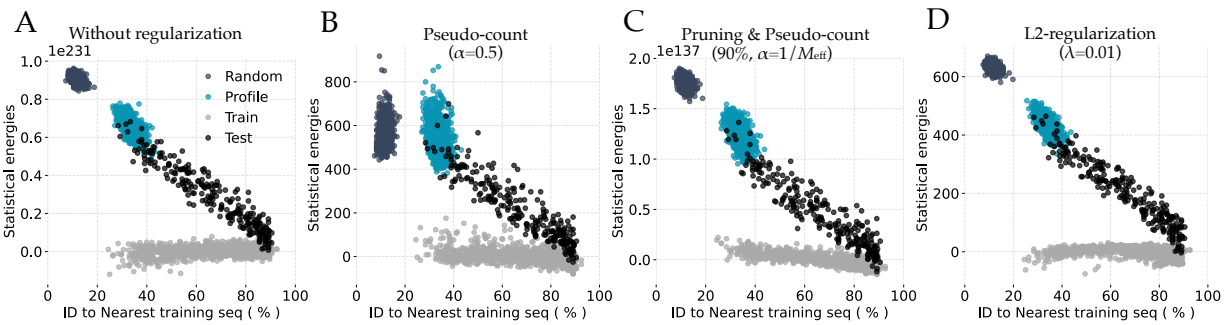

Figure 3.2: How Regularization Affects Statistical Energy Discrepancies
Statistical energies of training and test sequences against their ID to the Nearest training sequence for models trained with different regularization schemes. For all the figures the colors are the same: Training sequences in grey, Test sequences in black, Random sequences in dark blue and Profile sequences in blue. The training set was constructed by randomly selecting 1,007 sequences, while the remaining 251 were assigned to the test set. As a result, the test set includes sequences that are nearly identical to at least one sequence in the training set. The profile sequences correspond to the natural alignment where the columns have been randomized to erase the correlation signal. Please note that the y-axis scales are different in each panel. It is worth noting that the regularization parameters shown in each panel were selected following typical practices in the literature and are therefore not directly comparable.
A. **Without regularization**: Many parameters diverge, resulting in statistical energies reaching extreme values, up to 10^{231}. As a consequence, nearly all test sequences—except those that closely resemble a training sequence—are assigned significantly higher energies than the training data.
B. **Pseudo-count**: With $\alpha = 0.5$, the energy gap between test and training sequences is noticeably reduced. A similar trend is observed for profile and random sequences, whose energies also come closer to those of the natural sequences. For $\alpha = 1$ (not shown), these distinctions largely vanish, with all sequence types converging toward comparable energy levels.
C. **Pruning & Pseudo-count**: The pruning procedure is applied prior to inference, affecting only the couplings. Specifically, only the top 10% of pairwise frequencies are retained; the rest are set to zero. A small pseudo-count ($\alpha = \frac{1}{M_{\text{eff}}} \sim 0.001$) is also used to avoid parameter divergence.
D. **L_2-regularization**: The model is trained with a regularization strength $\lambda = 0.01$.
In both cases, regularization helps narrow the energy gap between training and test sequences. However, this gap remains substantial, particularly for test sequences that are more divergent from the training set.

As illustrated in Figure 3.2, all regularization strategies help reduce the discrepancy between the statistical energies of training and test sequences, primarily by preventing parameter divergence.

In practice, strong pseudo-counts have frequently been used in Mean Field DCA models for tasks such as contact prediction or mutational effect estimation ($\alpha = 0.5$ [38, 75]). This approach modifies empirical frequencies to avoid extreme parameter values, thereby reducing the energy gap between training and test sequences. However, at this level of regularization, random, profile-based, and some test sequences exhibit similar statistical energies (see Figure 3.2B).

DCA models inferred through maximum likelihood are often regularized using alternative strategies. Some studies combine weak pseudo-counts[2] with pruning procedures, which on their own fail to sufficiently constrain parameter divergence [59, 60]. Nevertheless, L_2 regularization remains the most common approach [30, 34, 43, 58, 77, 104, 107].

As shown in Figure 3.2C and D, these methods effectively reduce the energy gap between training and test sequences[3], thereby improving generalization. However, this comes at the cost of diminishing the contrast between natural sequences and those generated from random or profile models. As is often the case in statistical modeling, adjusting the regularization strength involves balancing generalization (variance) against model bias.

In the following, we will primarily focus on L_2 regularization, which is the most widely used technique for the inference of BM models. Unless explicitly stated, BM will thus refer to models inferred under this regularization scheme.

2: Typically $\alpha \sim \frac{1}{M_{\text{eff}}}$. M_{eff} being the effective number of sequences.

3: Though a substantial discrepancy remains, especially for test sequences that are more divergent from the training set.

3.2.1 L_2 regularization

From a Bayesian perspective, L_2 regularization introduces a zero-mean Gaussian prior over the model parameters. Increasing the regularization strength corresponds to reducing the prior variance, thereby pulling parameter values closer to zero and increasing bias.

In the context of BM models, this regularization strategy modifies the consistency Equation 2.6 and Equation 2.7 as follows:

$$f_i(a) = \sum_{\sigma} P(\boldsymbol{\sigma})\delta(\sigma_i, a) + 2\lambda_h h_i(a), \tag{3.2}$$

$$f_{ij}(a,b) = \sum_{\sigma} P(\boldsymbol{\sigma})\delta(\sigma_i, a)\delta(\sigma_j, b) + 2\lambda_J J_{ij}(a,b) \tag{3.3}$$

where λ_h and λ_J denote the regularization strengths for the fields and couplings, respectively.

Applying such regularization requires choosing suitable values for the hyperparameters λ_h and λ_J. For specific tasks, such as protein contact or mutational effect prediction, the regularization strength can be optimized with the goal of maximizing agreement with experimental data. Indeed, even if the model is not trained in a supervised way, the output can be directly compared to a known ground truth, and the regularization can be adjusted to maximize performance.

However, we are interested in the generative capacities of such models. In this context, it is not evident that the regularization strength yielding the best results for a specific task would also be optimal for generation. For instance, if the goal is to generate functional proteins, there is no fundamental reason to expect that the strongest inferred couplings should exclusively correspond to direct physical contacts in the protein's 3D structure.

As outlined in Section 2.3, evaluating generative models is inherently difficult and selecting an appropriate regularization strength is therefore equally non-trivial. In what follows, we analyse how varying the regularization strength shapes the parameters inferred by the model.

3.3 Undersampling-induced biases

As discussed in Section 1.5.1, proteins are characterized by networks of interdependent residues, with interactions occurring across multiple scales including:

- Pairwise local interactions that typically correspond to residues that are in contact within the three-dimensional structure.
- Networks of collective interactions that are essential for protein folding and function, including binding affinity, enzymatic catalytic activity [19], and allosteric communication [36][4]. These groups of coevolving residues are often referred to as Sectors in the literature [19, 29].

4: Allostery refers to the thermodynamic coupling of distant sites without direct physical interaction.

We hypothesise that a generative model should account for these interactions spanning multiple scales, from simple pairwise couplings to higher-order collective effects. In the following section, we review key results from Kleeorin *et al.*[30], who demonstrated that, in the undersampling regime typical of protein sequence data, BM models are unable to reliably infer these diverse interaction patterns. Their study

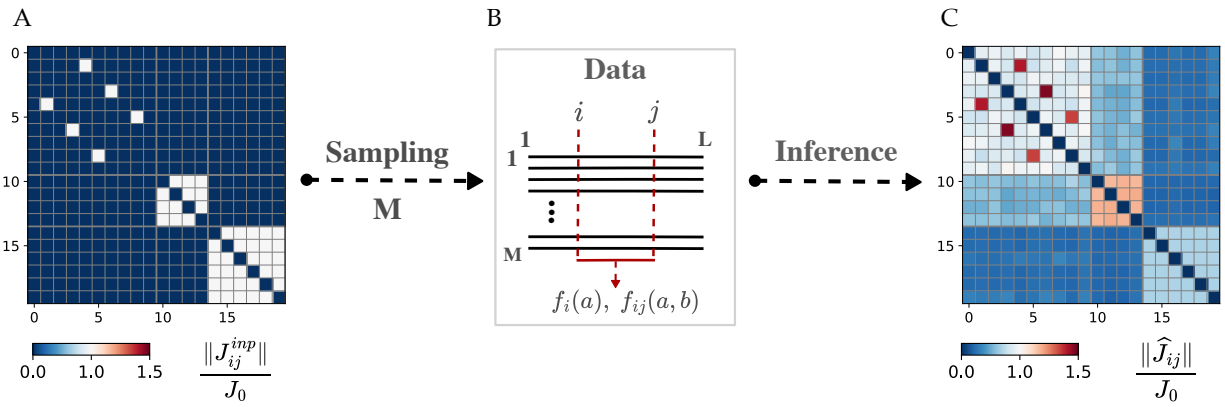

Figure 3.3: Inference from a toy Model.
The Toy model proposed by Kleeorin *et al.* [30] aims to capture a fundamental characteristic of proteins: the interaction of residues at various scales. The model has $L = 20$ positions, and $q = 10$ possible amino acids at each site.
A. The $L \times L$ Frobenuis norm of the input couplings with three types of features: isolated pairwise couplings in positions (2,5), (4,7) and (6,9), a small collective group in positions (11-14) and a large collective unit in positions (15-20). The couplings are normalized by the Frobenius value in the Zero-sum gauge.
B. In order to replicate the undersampling observed in natural protein families, $M = 300$ sequences are sampled from the input model and the one- and two-body frequencies are computed. Then the BM method is used to infer back the toy model from the M sampled sequences.
C. Frobenius norm of inferred couplings normalized by the input couplings.

was conducted with the pseudo-likelihood maximization DCA method (plmDCA) that approximates the likelihood by considering each position conditionally on all others. Here, we provide new results obtained with the standard BM approach (see Section 2.1.2), that are very similar to those presented in [30]. As in the original work, our analysis includes both synthetic models and real protein sequence data.

3.3.1 Toy Model

As discussed in the previous part, evaluating generative models is inherently challenging. An interesting strategy involves assessing model performance on synthetic data generated from a controlled toy model designed to capture key features of the system under study. The toy model proposed by Kleeorin *et al.* was chosen to capture key characteristics of real proteins, in particular, the presence of residue interactions at multiple scales.

The probability distribution of this toy model is given by Equation 3.1 with parameters h^{inp} and J^{inp}. All fields are set to zero, while the couplings are specified as illustrated in Figure 3.3A. The isolated pairwise couplings at positions (2,5), (4,7), and (6,9) represent contacts within the 3D structure. The two interconnected clusters at positions (11–14) and (15–20) correspond to groups of coevolving residues, analogous to sectors. The non-zero couplings in the model are set to $J^{inp} = 2$, and all remaining couplings are fixed at zero.

The task is to recover the original coupling patterns by training models on sequences generated from this toy model. Performance is assessed by calculating the Frobenius norm $\|\widehat{J}_{ij}\|$ of the inferred couplings[5] in the zero-sum gauge[6] and comparing these values to the input Frobenius value J_0. A perfect inference would correspond to $\|\widehat{J}_{ij}\| = J_0 = 6$ for all positions with non-zero couplings. In the following figures, the reported values correspond to an average over each type of couplings.

5: The frobenuis norm is computed as follows: $\|J_{ij}\| = \sqrt{\sum_{a,b} J_{ij}(a,b)^2}$.

6: Fields and Couplings are modified so that: $\sum_a h_i(a) = \sum_a J_{ij}(a,b) = 0$. See Section A.1 for details.

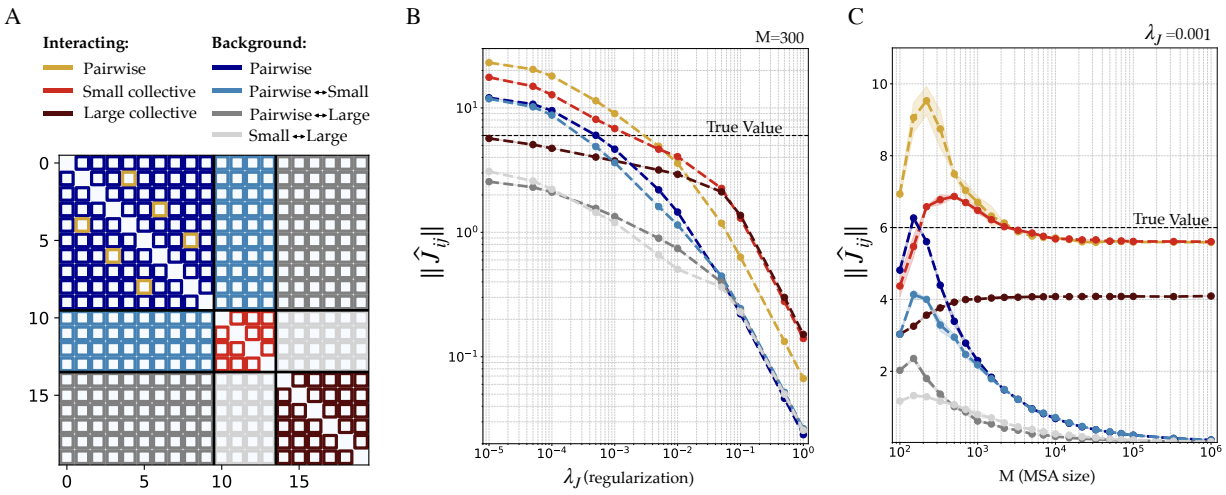

Figure 3.4: Inference from a toy Model.
The Toy model proposed by Kleeorin *et al.* [30] aims to capture a fundamental characteristic of proteins: the interaction of residues at various scales. The model has $L = 20$ positions, and $q = 10$ possible amino acids at each site.
A. $L \times L$ matrix showing the type of interaction in which each residue is involved: isolated pairwise couplings in positions (2,5), (4,7) and (6,9), a small collective group in positions (11-14) and a large collective unit in positions (15-20).
B. Magnitude of the inferred couplings as function of regularization. Contacts are the top couplings in the low-regularization limit while collective units are emphasized at high regularization.
C. Magnitude of the inferred couplings as function of MSA size for models inferred from the Toy Model. Regularization is set to $\lambda_h = 0.01$, $\lambda_J = 10^{-3}$. For low MSA size we observe sharp increase in the inferred couplings for pairwise couplings and non interacting couplings. The peak is less pronounced for the collective features.

To mimic the undersampling regime seen in natural protein families, we use a MCMC algorithm to draw $M = 300$ sequences from the input toy model that contains $\sim 10^4$ parameters[7]. Then, BM models are inferred from these samples, varying the regularization strength.

7: In this case, 14% of the amino acid combinations are absent from the training dataset.

Figure 3.4B illustrates how the different types of couplings are inferred as a function of the L_2 regularization strength. Given the undersampling regime, it is not possible to accurately recover all parameters. For example, spurious couplings between non-interacting residues are expected to arise. However, Figure 3.4B also highlights specific biases in the inference of pairwise and collective interactions.

At low regularization, pairwise features are overestimated relative to small collective features, which in turn are overestimated compared to large collective couplings. With increasing regularization strength, this trend reverses, and the inferred couplings are progressively dampened by the L_2-penalization. This behavior closely matches that described in the Kleeorin *et al.* study [30].

Unlike the original study, we also chose to separate the contributions associated with different types of non-interacting couplings. This allows us to show that these couplings are not affected in the same way by the undersampling regime. Couplings located in the background of the pairwise couplings, as well as those situated between the pairwise couplings and the small group, are the most strongly overestimated at low regularization. Their strength even exceeds that of the couplings associated with the large group. However, non-interacting couplings involving a residue from the large group remain significantly weaker than interacting couplings of the large collective feature.

Another analysis of coupling inference as a function of MSA size reveals an interesting phenomenon. In Figure 3.4C, we observe pronounced peaks in the Frobenuis norm of inferred couplings, which occur at

distinct MSA sizes depending on the feature type. As demonstrated in the original study by Kleeorin *et al.*, both the position and magnitude of these peaks are influenced by the nature of the couplings and the strength of the regularization.

Such behavior is reminiscent of the double descent effect, well-documented in supervised learning and characterized by sharp increases in generalization error when model complexity approaches the number of training examples.[8]

8: A dedicated study of this effect is currently being conducted by a master's student in our team, Milo Repossi.

The key result we highlight here is that there exists a systematic bias between the estimation of collective modes and isolated interactions. While varying the regularization strength can modulate this imbalance, it cannot fully cancel the bias.

3.3.2 Real data

To demonstrate that the results obtained with toy model-generated data are applicable to real data, Kleeorin *et al.* [30] have also examined coupling inference in models trained on the AroQ family of Chorismate Mutase (CM) enzymes. As discussed in [30], this protein family displays a pattern of correlations involving features of various scales, reminiscent of the toy model features. By distinguishing between couplings corresponding to direct structural contacts and those associated with functional sites, we can compare the inference of local versus collective interactions as a function of the regularization strength.

In Figure 3.5, we report the behavior of couplings associated with contacts, functional positions, and other interactions (as defined by Kleeorin *et al.*[9]) as a function of L_2 regularization. Here, we observe a pattern remarkably similar to that of the toy model: at low regularization, structural contacts are emphasized over functional positions, while increasing the regularization strongly dampens all inferred couplings and ultimately reverses their relative importance at high regularization.

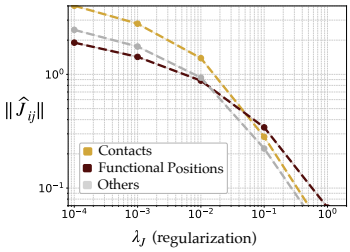

Figure 3.5: CM family: inferred couplings as function of L_2 regularization strength.
Inferred couplings as function of regularization in the CM family. The graph displays the average magnitude of inferred couplings for three groups of residues: functional positions (red), direct contacts (yellow), and all other position pairs (grey). Similar to what is observed in toy models, these results indicate that features of different effective sizes differentially dominate the inference as the regularization strength is increased.

In contrast to the toy model setting, real protein families lack a ground truth, making it impossible to directly compare inferred couplings to true parameters. Still, the results suggest that the inference biases observed in the toy model carry over to real data. In particular, weak regularization tends to highlight local, small-scale interactions, while stronger regularization emphasizes larger-scale features which are functionally essential positions in Chorismate Mutase.

9: For contacts, this corresponds to the *top pairs obtained in the low-regularization limit with APC that are in the contact map, but are not part of the functional network positions*. Functional positions, in contrast, are defined as the *top pairs obtained with strong regularization without APC that are in the functional network, but are not contacts* [30]. A pair is considered to be in the contact map if the two residues have at least one pair of atoms closer than 5 Å in the crystal structure of *E. coli* CM, excluding pairs that are less than four positions apart along the linear sequence.

3.4 Generative capacity of the Boltzmann Machine

The generative capacity of the Boltzmann Machine has been experimentally tested by Russ *et al.* [58] in 2020. In particular, they used the bmDCA method[10] with L_2 regularization to study the AroQ family of Chorismate Mutase (CM). In the following sections, we will briefly introduce this protein family and the workflow used to experimentally test these sequences.

10: bmDCA refers to the standard BM method, which uses gradient descent to minimize the negative log-likelihood.

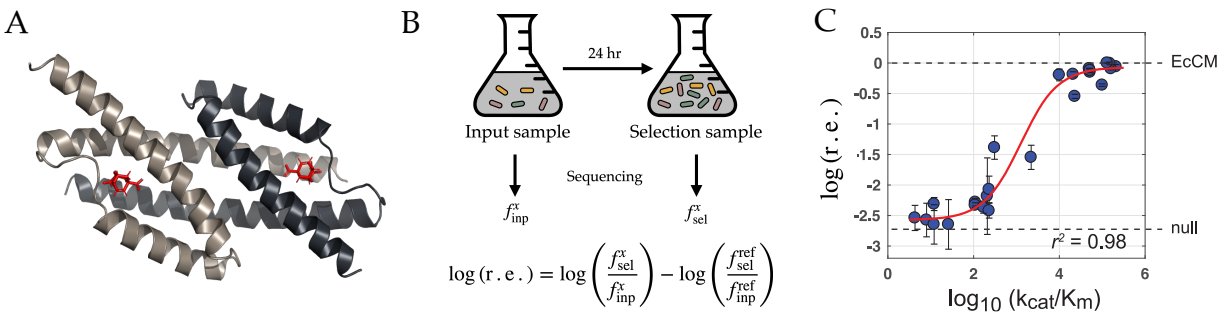

Figure 3.6: Experimental testing of Chorismate Mutase sequences. (Pannel C adapted from [58])
A. Ribbon representation of *E. coli* CM domain structure (PDB: 1ECM). AroQ CMs are dimers with two symmetric active sites. A bound substrate analog is shown in red.
B. Workflow for the experimental testing of CM sequences. CM-deficient E. coli cells carrying libraries of variants are grown under selective conditions, after which deep sequencing of input and selected populations is performed. The log (r.e.), i.e. logarithm of the relative enrichment is computed from the frequencies of a variant in the input and selected libraries compared to a reference.
C. The relationship of log (r.e.) to catalytic activity log (k_{cat}/K_m) for a number of CM variants yields a "standard curve." This shows that the log (r.e.) metric is indeed related to the activity of the CM sequences.

3.4.1 Testing of artificial CM sequences

The chorismate mutase family is a classical model system for studying the principles of catalysis and enzyme design [108]. These proteins are involved in the biosynthesis pathway for two amino acids: tyrosine (Tyr) and phenylalanine (Phe).

11: See Section 5.1.3 for further details.

The catalytic activity of enzymes, i.e., their capacity to accelerate specific chemical reactions, can be measured experimentally[11]. Yet, such assays are difficult to scale, limiting their use in evaluating large numbers of variants.

The role of the *E. coli* CM is essential for supporting the growth of the cell in a medium lacking Tyr and Phe. This allows the catalytic activity of CM sequences to be inferred indirectly, using the workflow illustrated in Figure 3.6B: (1) *E. coli* cells in which the native gene encoding the CM protein has been knocked out are engineered to express sequences from the library being tested; (2) these cells are then cultivated in a medium that requires them to express functional CM proteins in order to grow, serving as a selection process; (3) deep sequencing methods are used to determine the frequency of each variant before and after selection.

The metric used to distinguish functional variants from non-functional ones is the logarithm of the relative enrichment:

$$\log(\text{r.e.}) = \log\left(\frac{f^x_{sel}}{f^x_{inp}}\right) - \log\left(\frac{f^{ref}_{sel}}{f^{ref}_{inp}}\right) \tag{3.4}$$

where f^x_{inp} and f^x_{sel} are the frequencies of variant x before and after selection, respectively. These frequencies are compared to those of a reference sequence, here *E. coli* CM. As shown in Figure 3.6C, this metric has been validated as a reliable indicator of the catalytic activity for the tested enzymes.

Using this approach, Russ *et al.* systematically tested 1,130 natural sequences and a diverse set of artificial variants generated with various models.

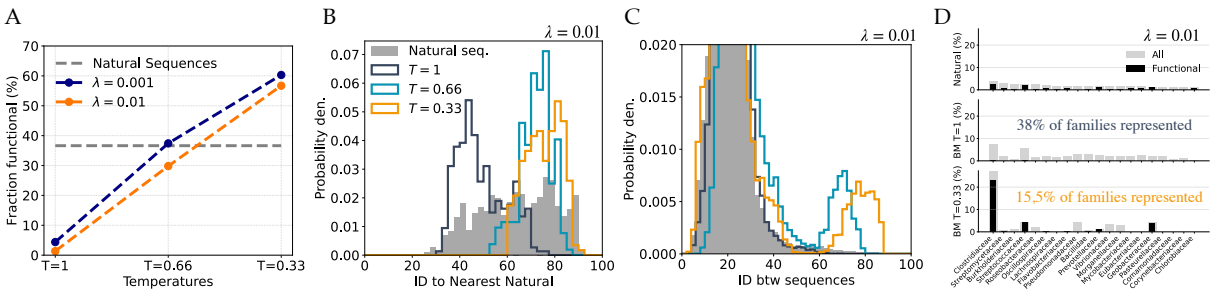

Figure 3.7: CM family: Artificial sequences as function of sampling temperature. (Figures made from Russ *et al.* data [58])
A. Fraction of functional sequences as function of temperature (for 2 regularization strengths: $\lambda = 0.01$ in orange and $\lambda = 0.001$ in blue) compared to natural sequences. At $T = 0.33$, the fraction of artificial sequences that are functional exceed the fraction of functional sequences in the natural set.
B. Histograms showing sequence identity to the Nearest Natural sequence across different sampling temperatures. Lower temperatures yield artificial sequences that more closely resemble natural ones.
C. Histograms of % of identities (ID) between sequences for the set of natural and artificial sequences generated at 3 different temperatures ($T = \{1, 0.66, 0.33\}$). As the sampling temperature decreases, the distribution deviates from that of natural sequences, resulting in an increasing number of sequence pairs with identity exceeding 60%.
D. The top bar plot displays the proportion of natural sequences belonging to the 20 most represented families in the natural alignment. For comparison, analogous statistics are shown for artificial sequences generated using a BM at T=1 and T=0.33; in these cases, each artificial sequence is assigned to the family of its closest natural counterpart. Notably, lowering the temperature induces a bias towards certain families, and the high proportion of functional sequences at T=0.33 can be explained by this effect.

3.4.2 Temperature rescaling

In this section, we present several analyses made from data provided by Russ *et al.* in their study [58], extending beyond the scope of their original work.

The fraction of functional sequences for the different datasets is displayed in Figure 3.7A. The first point to highlight concerns the natural sequences: among the 1,130 sequences tested, only 37% are functional under the experimental conditions described above. Therefore, the percentage of functional sequences generated by a model should always be interpreted in light of this reference value.

For two Boltzmann Machines (BM) regularized with L_2-penalties of 0.01 and 0.001, the percentage of functional sequences is 1.4% and 4.4%, respectively. This low fraction is attributed to a discrepancy between the statistical energies of the artificial and natural sequences, largely driven by the applied regularization.

To generate sequences with lower energies, an additional hyperparameter was thus introduced: the temperature. The models used to sample such sequences are thus defined as:

$$P(\{\sigma_i\}_{i=1,\dots,L}) \sim e^{-\frac{E(\boldsymbol{h},\boldsymbol{J})}{T}}, \tag{3.5}$$

with $T = \{1, 0.66, 0.33\}$. Sampling at these temperatures effectively rescales all parameters and reduces the energies of the generated artificial sequences.

As shown in Figure 3.7A, lowering the sampling temperature results in a significantly larger fraction of functional artificial sequences. However, this gain in functionality comes at the cost of reduced diversity and novelty, as illustrated in Figure 3.7A and B. In other words, the model does not only generate sequences closer to natural proteins; it also tends to produce sequences that are more similar to one another[12].

This is not a minor observation. Notably, the fraction of functional sequences at $T = 0.33$ exceeds that of natural sequences. This suggests

12: In Appendix B we provide supplementary figures showing that a similar behavior can be observed with the toy model introduced in Section 3.3.1.

that the model is biased toward sequences that remain functional under the chosen experimental conditions. In Figure 3.7C, we display the percentage of sequences belonging to the 20 most represented families[13] in the alignment, along with the fraction of functional sequences within each family. When we compute the same statistics for artificial sequences, by assigning each one to the family of its closest natural neighbor, we observe that lowering the temperature indeed introduces a bias toward certain families. This bias in family representation helps explain the high proportion of functional sequences observed at $T = 0.33$[14].

These findings highlight the potential of the BM as a generative model for protein sequences, but also underscore the need for alternative inference strategies to correct the biases identified by Kleeorin *et al.* and avoid relying on low-temperature sampling.

The next chapter thus introduces a new approach to overcome this undersampling problem in generative models for protein sequences: the Stochastic Boltzmann Machine (SBM).

13: In biological taxonomy, a family is a rank that groups together several closely related genera that share key anatomical and evolutionary traits. A genus (plural: genera), in turn, encompasses one or more species that are even more closely related to each other.

14: It is worth noting that not all families are represented in the model at $T = 1$ either. However, this result should be interpreted with care, as it also reflects the fact that the amount of artificial data is here smaller than the number of families in the natural dataset. In the following chapter, we show that using a BM model inferred with $\lambda = 0.01$, the percentage of families represented, calculated over 10,000 sequences, rises to more than 93% at $T = 1$. The complete histograms are provided in Appendix B and Appendix C, along with the proportions of families present across the different artificial sequence datasets. We also provide the same analysis for the model inferred with $\lambda = 0.001$ for which artificial sequences are even closer to the natural ones.

Overcoming undersampling-induced biases

4

Following the previous chapter, we introduce here the Stochastic Boltzmann Machine (SBM) algorithm, designed to address the undersampling-induced biases in generative models for protein sequences.

This project was carried out in collaboration with members of the Ranganathan laboratory at the University of Chicago. In particular, Yaakov Kleeorin was the first to propose the SBM method and to initiate its development.

The original goal of my PhD was to investigate generative models for bacterial genomes using the energy-based models discussed throughout this thesis. However, as we will discuss in this chapter, the SBM approach and its application to protein sequences proved rich and worthy of deeper investigation.

In this chapter, we will first discuss some analyses that I conducted using the toy model already presented in Section 3.3.1. Then we will move onto real data with the chorismate mutase protein family, a commonly studied system already introduced in the previous chapter[1].

1: The code used to produce the results presented in this work is available at: https://github.com/marionchv/SBM.git

The theoretical work presented here is complemented by experimental results obtained by Emily Hinds, a PhD student working in Rama Ranganathan's group.

4.1 Navigating the parameter space of a toy model

Our goal is to infer the parameters of the Boltzmann distribution defined in Equation 3.1 by maximizing the log-likelihood function.

In what follows, we thus consider a gradient descent applied to an objective function defined as the negative log-likelihood $-\mathscr{L}(\boldsymbol{h}, \boldsymbol{J})$, augmented with L_2-regularization terms:

$$f(\boldsymbol{\theta} = \{\boldsymbol{h}, \boldsymbol{J}\}) = -\mathscr{L}(\boldsymbol{h}, \boldsymbol{J}) + \lambda_J \|\boldsymbol{J}\|^2 + \lambda_h \|\boldsymbol{h}\|^2. \tag{4.1}$$

As mentioned in Chapter 2, the parameters $\boldsymbol{\theta} = \{\boldsymbol{h}, \boldsymbol{J}\}$ are updated via gradient descent according to:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \eta_t \boldsymbol{p}_t \tag{4.2}$$

where $\boldsymbol{p}_t$ denotes the search direction, η_t the learning rate, and $\boldsymbol{\theta}_t \in \mathbb{R}^{N_p}$ a vector containing all model parameters at step t. In Figure 4.1, we recall several key hyperparameters of the optimization procedure: the number of gradient descent steps (N_{iter}), the number of Monte Carlo steps (k_{mc}), and the number of generated samples used to compute single-site and pairwise frequencies with respect to the model (N_{chains}).

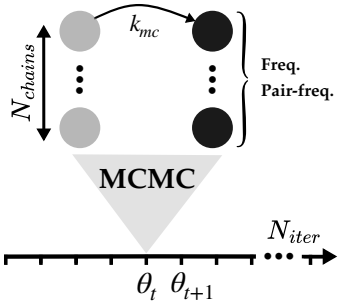

Figure 4.1: Schematic illustration of the gradient descent algorithm.
The model parameters at iteration t of the gradient descent are stored in the vector $\boldsymbol{\theta}_t$. At each step, a Monte Carlo simulation is used to sample N_{chains} sequences from the model. The sequences obtained after k_{mc} MCMC steps are then used to compute single-site and pairwise frequencies, which are required to evaluate the gradient of the objective function.

The search direction $\boldsymbol{p}_t$ and the resulting learning dynamics depend on the specific optimization algorithm employed. In the following, we briefly review two optimization algorithms: the Vanilla Gradient Descent method (VGD) and a quasi-Newton method known as L-BFGS. Both are

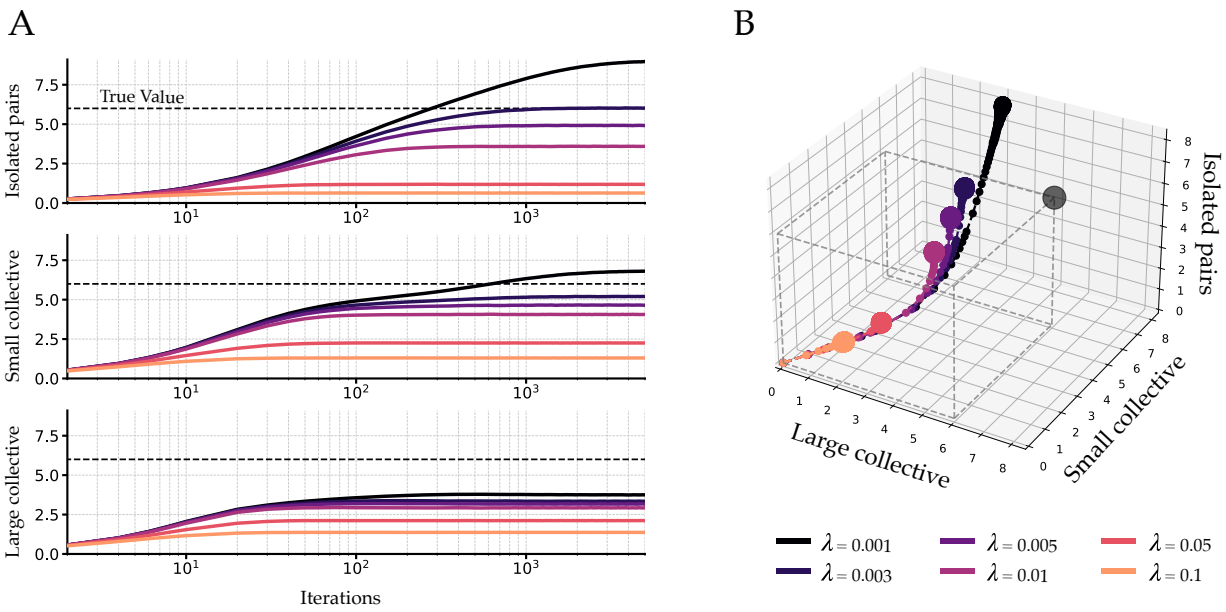

Figure 4.2: Inference of toy model couplings as a function of gradient descent iterations for the BM^{VGD} method.
For all these panels, the magnitude of the couplings are computed from the Frobenius norm of the coupling matrix, by averaging over the different types of interacting couplings. These models were inferred under varying levels of L_2-regularization. In all cases, the inference was initialized with parameters set to zero.
A. Magnitude of the inferred couplings over iterations for different L_2-regularization strengths. The true values of the toy model couplings are indicated by a dotted black line. The three panels correspond to the three types of couplings: isolated pairs (top), small collective groups (middle), and large collective groups (bottom).
B. Using the same data as in panel A, we show the inference trajectories in a 3D space defined by the isolated, small collective, and large collective couplings. Larger markers indicate that successive points in the trajectory are close in parameter space, reflecting a slowdown in the inference dynamics. The grey dotted cube indicates the target values in each direction, and the grey dot corresponds to the point where all interacting couplings are correctly inferred.

applied to the toy model introduced in Section 3.3.1, which provides a controlled framework to explore the undersampling induced biases and to motivate the use of the L-BFGS algorithm. Unless otherwise stated, all models are inferred from a dataset of 300 sequences generated with the toy model.

4.1.1 Vanilla Gradient Descent

In the Vanilla Gradient Descent algorithm, the search direction is determined by the gradient of the objective function:

$$p_t = -\nabla f(\theta_t). \tag{4.3}$$

However, relying exclusively on gradient information to navigate the parameter space is suboptimal, especially when the curvature is highly heterogeneous. Since curvature is characterized by the Hessian matrix, strong inhomogeneity corresponds to Hessian eigenmodes spanning several orders of magnitude. In the case of a Maximum Entropy model, the Hessian is equal to the susceptibility matrix associated with the statistical constraints defining the model [109]. This matrix notably contains correlations, which, at least near convergence, should reflect the multiscale correlations present in natural data.

It is thus reasonable to expect that the high-dimensional parameter space explored during optimization exhibits highly anisotropic curvature: some directions may be steep, while others are relatively flat.

In Figure 4.2, we analyze the inference of isolated, small collective, and large collective couplings as a function of the number of VGD iterations. The method combining the Boltzmann Machine and Vanilla Gradient Descent is hereafter referred to as BM^{VGD}.

As previously shown in Section 3.3.1, the inferred parameters after 5000 iterations are biased. Specifically, Figure 4.2A shows that with weak regularization ($\lambda = 0.001$), pairwise couplings are overestimated, whereas large collective couplings are underestimated. A moderate increase in regularization ($\lambda = 0.003$) improves the inference of pairwise interactions, though collective ones remain poorly estimated. With strong regularization ($\lambda = 0.1$), the balance shifts in favor of collective couplings, but all interactions end up being substantially underestimated.

In Figure 4.2C, the dynamics as function of iterations are visualized directly in the 3D space of the three coupling types. Interestingly, we observe that the initial inference trajectories are nearly identical across different regularization strengths.

Then, when L_2-regularization is not too strong, the trajectories begin to separate once the large collective couplings converge and the inference of pairwise couplings accelerates.

It is not surprising that these different types of couplings exhibit different learning dynamics. For example, a recent study by Catania *et al.* [110] examined the evolution of the eigenmodes of the coupling matrix during the inference of an Ising model (using a mean-field approximation) and observed that different modes do not have the same dynamics.

Nevertheless, it is important to emphasize that the observed dynamics depend on the sampling regime of the training data. For instance, we provide additional figures in Section C.2 showing that under full sampling conditions, the learning dynamics of BM^{VGD} differ substantially.

Beyond the suboptimality of VGD, what matters here is that the trajectory of the inference in parameter space does not allow for accurate recovery of the true parameters under these undersampling conditions[2].

2: Note that this trajectory is independent of the hyperparameter N_{chains}; see Figure C.2 in Appendix C.

4.1.2 Quasi-Newton methods

The DCA method has been implemented in various forms, several of which rely on quasi-Newton algorithms that significantly accelerate gradient descent convergence [62, 111, 112].

In quasi-Newton gradient descent, the search direction is defined as:

$$\boldsymbol{p}_t = -\boldsymbol{B}_k^{-1} \nabla f(\boldsymbol{\theta}_t) \tag{4.4}$$

where $\boldsymbol{H}_k = \boldsymbol{B}_k^{-1}$ is an approximation of the inverse Hessian matrix.

One of the most widely used quasi-Newton algorithms is the BFGS method, which we briefly describe below.

4.1.3 BFGS algorithm

In the BFGS method [113], named after Broyden, Fletcher, Goldfarb, and Shanno, the inverse Hessian approximation is iteratively updated

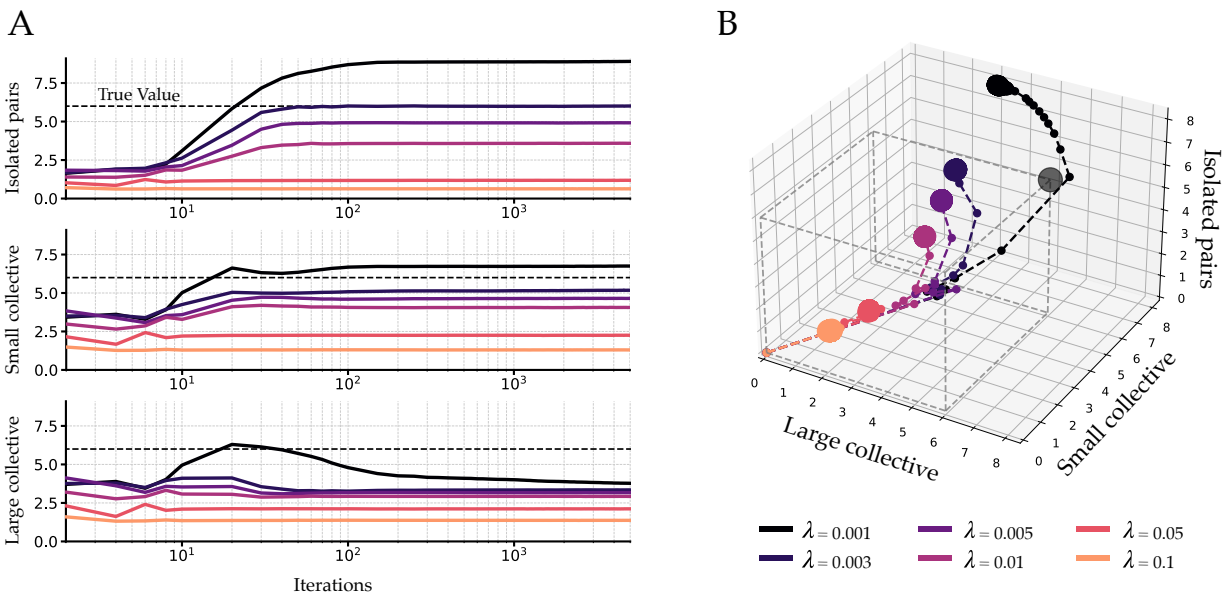

Figure 4.3: Inference of toy model couplings as a function of gradient descent iterations for the BM$^{\text{LBFGS}}$ method.
For all these panels, the magnitude of the couplings are computed from the Frobenius norm of the coupling matrix, by averaging over the different types of interacting couplings. These models were inferred under varying levels of L_2-regularization. In all cases, the inference was initialized with parameters set to zero.
A. Magnitude of the inferred couplings over iterations for different L_2-regularization strengths. The true values of the toy model couplings are indicated by a dotted black line. The three plots correspond to isolated pairs (top), small collective groups (middle), and large collective groups (bottom).
B. Using the same data as in panel A, we show the inference trajectories in a 3D space defined by the isolated, small collective, and large collective couplings. Larger markers indicate that successive points in the trajectory are close in parameter space, reflecting a slowdown in the inference dynamics. The grey dotted cube indicates the target values in each direction, and the grey dot corresponds to the point where all interacting couplings are correctly inferred.

according to:

$$\boldsymbol{H}_t = \boldsymbol{V}_{t-1}^T \boldsymbol{H}_{t-1} \boldsymbol{V}_{t-1} + \frac{\boldsymbol{s}_{t-1}\boldsymbol{s}_{t-1}^T}{\boldsymbol{y}_{t-1}^T \boldsymbol{s}_{t-1}}, \tag{4.5}$$

where we introduce the matrix $\boldsymbol{V}_t = \mathbf{I} - \frac{\boldsymbol{y}_t \boldsymbol{s}_t^T}{\boldsymbol{y}_t^T \boldsymbol{s}_t}$, along with two standard quantities:

$$\boldsymbol{s}_t = \boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t \text{ and } \boldsymbol{y}_t = \nabla f(\boldsymbol{\theta}_{t+1}) - \nabla f(\boldsymbol{\theta}_t). \tag{4.6}$$

3: The performance of BFGS can degrade if the line search does not satisfy the Wolfe conditions [113].

This algorithm is known for its robustness and exhibits a superlinear rate of convergence, in contrast to the quadratic convergence of the Newton method[3].

However, each iteration requires $\mathcal{O}(N_p^2)$ arithmetic operations to compute the search direction, in addition to the cost of evaluating the gradient via Monte Carlo sampling. Given that $N_p \sim 10^5$–10^7, storing and updating a full Hessian matrix is computationally and memory-wise impractical.

To overcome this limitation, we now turn to the Limited-memory BFGS variant, which avoids the need to explicitly store and compute the full $N_p \times N_p$ matrix.

4.1.4 L-BFGS algorithm

The L-BFGS algorithm was first introduced by Nocedal in 1980 [114]. For comprehensive technical details, the reader is referred to the book *Numerical Optimization* [113]. Here, we briefly summarize the key concepts behind its implementation.

The central idea of L-BFGS is to use curvature information from the most recent iterations to construct an approximation of the inverse Hessian. Rather than storing a fully dense $N_p \times N_p$ matrix, the method retains only a small number of vectors of length N_p, which implicitly represent the approximation and make it particularly well suited for large-scale problems.[4]

4: It also does not require knowledge of the Hessian's sparsity structure or any separability in the objective function.

The algorithm is nearly identical to BFGS, with the main difference lying in how the inverse Hessian is updated. The user specifies the number of corrections m to retain and provides a positive-definite initial matrix $\boldsymbol{H}_0$. During the first m iterations, the updates are identical to those in BFGS. Then, for $k > m$, $\boldsymbol{H}_t$ is constructed by applying m BFGS updates to $\boldsymbol{H}_0$, using information from the m most recent iterations[5].

5: When $m = \infty$, the algorithm becomes equivalent to standard BFGS.

In our implementation, we rely on a simple and efficient algorithm [113] to directly compute the product $\boldsymbol{p}_t = -\boldsymbol{H}_t \nabla f(\boldsymbol{\theta}_t)$ from the m most recent pairs of vectors $\{\boldsymbol{s}_t, \boldsymbol{y}_t\}$ stored during optimization. Further technical details are provided in Appendix C.

Using this method with L_2-regularization, denoted BM^{LBFGS}, we now analyze the inference of the different types of couplings defined in the toy model as a function of iterations. The results are provided in Figure 4.3, for models inferred under the same undersampling and regularization conditions as previously.

We first observe that incorporating curvature information via L-BFGS significantly accelerates the convergence of gradient descent. However, our primary interest lies not only in convergence speed, but also in mitigating undersampling-induced biases.

Although the algorithm converges to the same biased solution as VGD (as expected, since the objective function remains unchanged), the trajectory followed in parameter space differs. Interestingly, in Figure 4.3B, we observe that for the least regularized model, the trajectory passes near the point corresponding to the true interacting couplings[6].

6: See Figure C.3 in appendix, showing that this is also the case for different values of the parameter m, and without L_2-regularization.

When examining the evolution of large collective couplings over iterations, we note a local optimum in their inferred magnitude around iteration 20. However, their intensity subsequently decreases and eventually converges to the same underestimated value as in the VGD case.

This peak coincides with a phase in which the isolated couplings begin to grow. In practical scenarios, where the ground truth is unknown, exploiting such transient behavior to stop inference at the right moment appears challenging. This suggests that at least some prior knowledge about the data structure may be necessary.

In summary, although the final biases persist, the trajectory observed in parameter space provides a potential opportunity to approach the true parameters and possibly mitigate undersampling-induced biases[7].

7: There is no clear evidence that this behavior could extend to other models or datasets. The present case involves a BM model with structured data such that different dimensions are affected unevenly by undersampling. The fact that the L-BFGS trajectory passes near the ground truth—even without knowing it a priori—is intriguing. To better understand this phenomenon and assess its generality, Milo Repossi, a student in our group, is currently studying inference trajectories in the case of a Gaussian model with different variance modes.

The next section introduces the Stochastic Boltzmann Machine, based on the L-BFGS algorithm and associated with an implicit regularization mechanism.

4.2 Stochastic Boltzmann Machine (SBM)

The goal here is to eliminate the need for explicit L_2-regularization by leveraging the hyperparameters of the L-BFGS gradient descent to induce an implicit regularization.

Among the different hyperparameters involved in the optimization, we focus here on the following:

- N_{iter}: the number of gradient descent iterations,
- N_{chains}: the number of Monte Carlo chains, i.e., the number of sequences used to compute model statistics at each gradient step,
- m: the number of stored vector pairs used to approximate the inverse Hessian. The curvature information used at each step then reflects only the m most recent updates.

Throughout this section, the regularization parameters λ_J and λ_h associated with the L^2 penalty terms are set to zero.

The divergence of parameters typically caused by undersampling will be controlled by: i) choosing a small value of m, which slows down the convergence of the L-BFGS algorithm; ii) limiting the number of iterations N_{iter} to take advantage of this slower convergence, a technique known in the machine-learning literature as early stopping [115]; and iii) using a small N_{chains} so that model statistics are themselves computed in an undersampled regime.

4.2.1 Slowing down convergence and leveraging MCMC noise for regularization

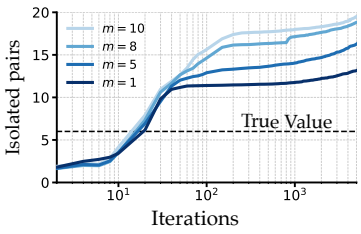

Figure 4.4: Convergence of L-BFGS as a function of m ($N_{\text{chains}} = 1000$). Magnitude of the inferred couplings for isolated pairs over iterations, for different values of m. The dashed black line indicates the true coupling value from the toy model. Reducing m slows convergence, and for $m = 1$, the algorithm exhibits a clear plateau. No explicit regularization is applied; with an infinite number of iterations, the optimization would eventually diverge.

Figure 4.4 shows the evolution of isolated pair couplings during the inference, using L-BFGS with different values of m.

The learning dynamics for these couplings appear to exhibit several distinct phases. The first phase, spanning roughly the initial 10 iterations, shows minimal progress—this behavior is also observed for the other types of couplings (see Appendix C). This is followed by a second phase of accelerated convergence, which is notably influenced by the choice of m: smaller values of m lead to an earlier termination of this acceleration phase. The learning dynamics then enter a plateau lasting until approximately iteration 1000, after which the inferred couplings begin to increase again. While only the first 5000 iterations are shown, the couplings continue to grow beyond that point due to the absence of regularization.

This plateau behavior is not specific to isolated pairs and is also observed for the other types of couplings (see Appendix C for supporting figures). Importantly, without prior knowledge of the data structure, we can track the average over all couplings and use this plateau as a practical stopping criterion. The objective is to stop the inference before the couplings begin to diverge again. In this example this corresponds to a range of approximately $N_{\text{iter}} \sim 100$–1000.

In addition to m, the plateau level is also determined by the number of Monte Carlo chains, N_{chains}. So far, all models have been inferred using $N_{\text{chains}} = 1000$, a setting in which isolated couplings remain systematically overestimated, even when $m = 1$.

To assess the impact of N_{chains}, we refer to Figure 4.5, which shows the evolution of these couplings in an undersampled setting, with $m = 1$ fixed and N_{chains} varied.

As the figure illustrates, adjusting this parameter has a clear effect on the plateau height. In particular, decreasing N_{chains} can be seen as increasing the effective strength of regularization, helping to mitigate the overestimation of isolated couplings.

It is not surprising that N_{chains} affects the optimization process, since both the gradient and Hessian are estimated from Monte Carlo (MCMC) samples. This approach is reminiscent of the philosophy of stochastic gradient descent (SGD) [116], in which model statistics are compared to empirical statistics computed on successive minibatches. However, here, we do not subsample the real data but rather control the fidelity of model statistics through intentional undersampling.

One might intuitively aim to match the degree of undersampling in the model statistics to that of the data. In practice, however, we find that more aggressive undersampling on the model side is required to achieve optimal regularization in this toy model setting.

For sufficiently low values of N_{chains} (~ 100), inference trajectories tend to slow down near the true value of the isolated couplings. At the same time, we observe significant variability between runs with identical N_{chains}, because we use the stochasticity inherent to MCMC to navigate the optimization landscape. In what follows, a single SBM model will thus refer to an average over multiple inference runs (typically 10).

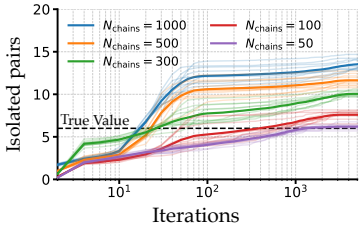

Figure 4.5: Convergence of L-BFGS as a function of N_{chains} ($m = 1$). Magnitude of couplings associated with isolated pairs during optimization, plotted as a function of iterations for various values of N_{chains}. Thin lines represent individual inference runs; the thick line indicates the average across runs. The dashed black line shows the true value from the toy model. Decreasing N_{chains} improves convergence toward this true value.

4.2.2 Mitigating Biases

We now return to the question of how this strategy affects the inference of other types of couplings beyond isolated pairs. Figure 4.6 shows the dynamics of isolated, small collective, and large collective couplings throughout training, for different values of N_{chains}, using the same layout as in earlier figures.

As illustrated in Figure 4.6A, lowering N_{chains} reduces the overestimation of pairwise couplings without significantly impairing the inference of collective interactions, in contrast to what is typically observed with L_2 regularization.

Then using this setting, we can access and stop closer to the true model parameters while being in an undersampling regime. To better compare BM and SBM, we present in the following section a systematic analysis of the two methods as a function of regularization strength.

4.3 Inference of Collective and Local Features

In this section, we systematically compare couplings inference on both the toy model and real protein sequences using the following approaches:

- BM^{VGD}: Inference using vanilla gradient descent[8], with $N_{\text{chains}} = 1000$, $k_{\text{mc}} = 10^5$, $N_{\text{iter}} = 3 \cdot 10^4$, L_2-regularization with $\lambda_h = 0.01$, and λ_J as a tunable parameter.
- SBM: Inference using the L-BFGS algorithm with $m = 1$, $k_{\text{mc}} = 10^5$, $N_{\text{iter}} = 400$. Each result is averaged over 10 independent runs, and N_{chains} is varied to control the strength of regularization.

8: As discussed in the previous section, the standard BM can be inferred using the L-BFGS algorithm, yielding similar coupling biases. We opt for vanilla gradient descent here, as L-BFGS may skip too many steps when L_2-regularization is very weak.

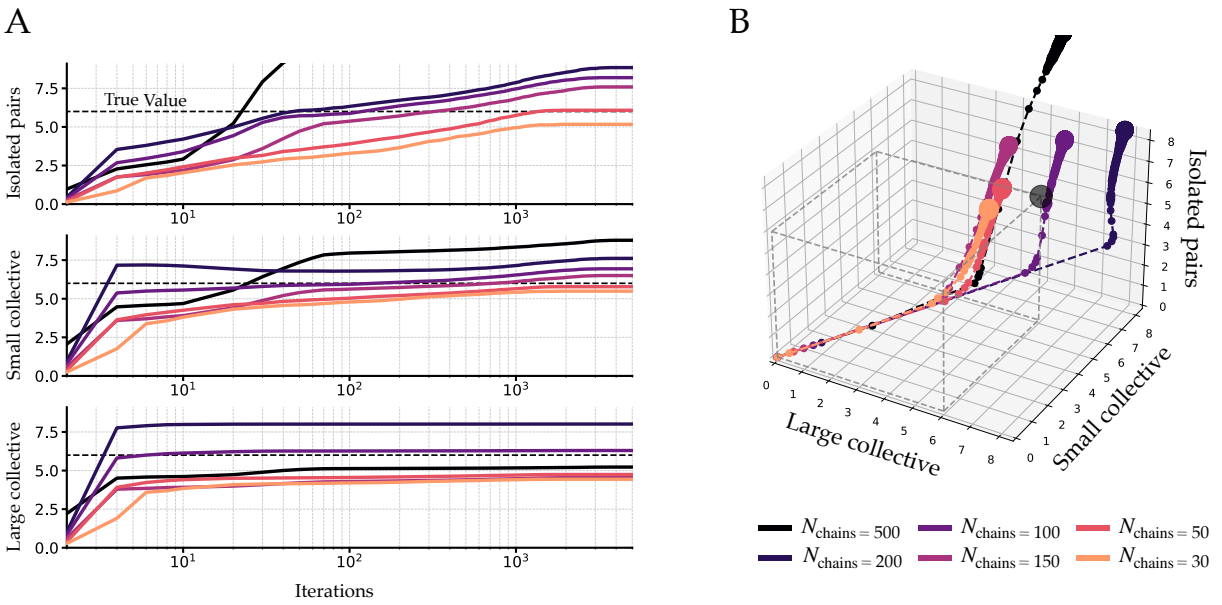

Figure 4.6: Inference of toy model couplings as a function of gradient descent iterations using the SBM method.
For all these panels, the magnitude of the couplings is computed from the Frobenius norm of the coupling matrix, averaged over the different types of interacting couplings. These models were inferred with different values of N_{chains}. In all cases, the inference was initialized with parameters set to zero.
A. Magnitude of the inferred couplings over iterations for different N_{chains}. The true values of the toy model couplings are indicated by a dotted black line. The three plots correspond to isolated pairs (top), small collective groups (middle), and large collective groups (bottom).
B. Using the same data as in panel A, we show the inference trajectories in a 3D space defined by the isolated, small collective, and large collective couplings. Larger markers indicate that successive points in the trajectory are close in parameter space, reflecting a slowdown in the inference dynamics. The grey dotted cube indicates the target values in each direction, and the grey dot corresponds to the point where all interacting couplings are correctly inferred.

4.3.1 Toy Model

As in the previous section, we investigate the behavior of the BM and SBM methods on synthetic sequences generated from the toy model proposed by Kleeorin *et al.* [30].

We consider the same undersampled regime as before, where inference is performed using 300 sequences generated from the toy model (see Figure 4.7A). We then compare the Frobenius norm of inferred couplings, averaged by coupling type, to the ground truth parameter values (namely, 6 for interacting couplings and 0 for non-interacting ones).

In Figure 4.7B and C, we show coupling inference as a function of the applied regularization: λ_J for the BM and N_{chains} for the SBM.

The biases exhibited by the BM model were discussed in Section 3.3.1; by contrast, the SBM behaves quite differently, as illustrated in Figure 4.7C.

For $N_{\text{chains}} \sim 1000$, we recover the same biases seen in the BM under weak regularization, namely an overestimation of isolated pairwise couplings and small collective interactions.

For very low values of N_{chains} (e.g., below 10), the bias reverses, as in the case of strong L_2 regularization in the BM, and isolated pairwise couplings become slightly weaker than collective ones. Furthermore, for models inferred with $N_{\text{chains}} < 10$, we observe greater variability across different inference runs, even when averaging over 10 models as done here.

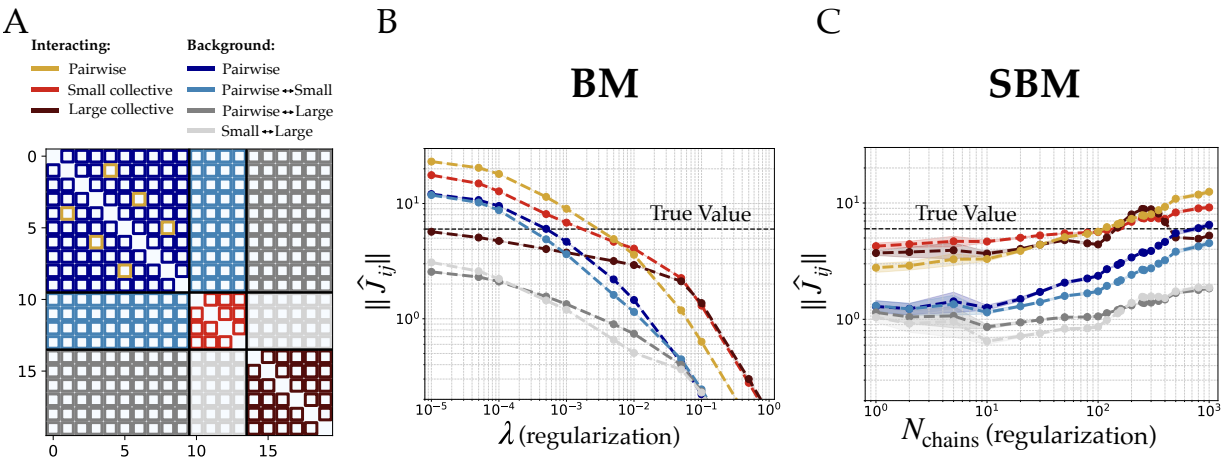

Figure 4.7: Toy model: BM and SBM inference as function of regularization.
The toy model proposed by Kleeorin *et al.* **[kleeorin2023undersampling]** is designed to capture a key feature of proteins: the presence of interactions at multiple spatial scales. The model includes $L = 20$ positions and $q = 10$ possible amino acids per site.
In panels B & C, each point is the mean of 10 independent repeats. BM: average over 10 independently inferred BMs. SBM: average over 10 SBMs, where each SBM is itself the average of 10 independently inferred models. Error bars represent the 5th–95th percentile interval obtained with the 10 repeats.
A. $L \times L$ matrix showing the interaction structure: isolated pairwise couplings at positions (2,5), (4,7), and (6,9) (yellow); a small collective group spanning positions 11–14 (light red); and a large collective unit from positions 15–20 (dark red). Non interacting positions are divided in four groups corresponding to the background of pairwise couplings (dark blue); the background between pairwise couplings and the small collective unit (light blue); the background between pairwise couplings and the large collective unit (grey); and the background between the two collective units (light grey).
B. BM model: the inference is done using vanilla gradient descent,with $N_{\text{chains}} = 1000$, $k_{\text{mc}} = 10^5$, $N_{\text{iter}} = 3 \cdot 10^4$, L_2-regularization with $\lambda_h = 0.01$, and λ_J as a tunable parameter. We show the magnitude of inferred couplings as a function of λ_J. In the low-regularization regime, isolated contacts dominate, while strong regularization emphasizes collective features.
C. SBM model: the inference is done with the L-BFGS algorithm with $m = 1$, $k_{\text{mc}} = 10^5$, $N_{\text{iter}} = 400$. Each result is averaged over 10 independent runs, and N_{chains} is varied to control the strength of regularization. We show the magnitude of inferred couplings as a function of N_{chains} (with large N_{chains} corresponding to weak regularization). A wide range of N_{chains} yields unbiased estimates of interacting couplings.

However, unlike the BM model, increasing the regularization of the SBM (i.e., decreasing N_{chains}) does not substantially reduce the magnitude of inferred couplings. More importantly, for intermediate values of N_{chains}, we observe a broad regime where interacting couplings are inferred with minimal bias and remain close to their true values. Finally, while non-interacting couplings are not driven to zero, their magnitudes remain lower than those of true interactions.

These encouraging results motivated us to apply the method to real data, focusing in particular on the Chorismate Mutase family.

4.3.2 Real Data

We now investigate the behavior of the two methods on real protein data. Specifically, we focus on the Chorismate Mutase family, introduced in Section 3.4.1.

The results of coupling inference for contacts and functionally relevant positions using the BM model have already been discussed in Section 3.3.2. While we no longer have access to a ground-truth model for comparison, we do observe trends that are consistent with those highlighted using the toy model, in agreement with the findings of Kleeorin *et al.* [30].

In Figure 4.8B & C, we compare these results with those obtained using the SBM model.

Using the SBM method, the relative importance of different coupling types as a function of N_{chains} follows a trend similar to that observed

with the BM as function of L_2 penalty: contact-associated couplings tend to have higher magnitudes at large N_{chains}, and this tendency reverses as regularization increases. However, as with the toy model, SBM regularization does not reduce coupling magnitudes as drastically as L_2-regularization does.

From these figures alone, we cannot draw conclusions about the generative capacity of the SBM model, nor can we identify an optimal level of regularization.

Nevertheless, it is worth noting that BM models inferred for this family have already been tested experimentally [58]. Thus, it is known that BM models trained with $\lambda = 0.01$ and $\lambda = 0.001$ can only generate less than 5% of functional sequences at T=1 (see Section 3.4.2).

For these models, artificial sequences sampled at $T = 1$ also exhibit, on average, higher statistical energies than natural sequences. Interestingly, the low-temperature sampling procedure required to obtain a higher fraction of functional sequences also induces a downward shift in the energy distribution of artificial sequences.

This phenomenon, originally reported by Russ *et al.* [58], is illustrated in Figure 4.8E for a BM model with $\lambda = 0.01$ [9]. In our case, sampling at $T = 0.75$ produces artificial sequences with average energies comparable to those of the training set [10].

As discussed in Section 3.4.2, low-temperature sampling is not a neutral operation, as it reduces the novelty and diversity of synthetic sequences. Regarding novelty, Figure 4.8D shows that sequences generated at $T = 0.75$ are indeed, on average, more similar to natural sequences.

Let us now consider an SBM model with a comparable novelty level. This is the case for the model inferred with $N_{\text{chains}} = 50$. For this model, Figure 4.8F shows that sequences generated at $T = 1$ have statistical energies similar to those of the training set.

Importantly, such result cannot be achieved with the BM approach. Indeed, BM models that produce artificial sequences with energies comparable to those of the training data at $T = 1$ require a level of regularization so low that most of the synthetic sequences would largely resemble the natural ones (see Figure C.8).

In contrast, the SBM model seems to bypass the need for low-temperature sampling [11]. But what does this imply, and does it make the SBM a better model? To address these questions, we now turn to the results of experimental assays performed on sequences generated by these models.

9: We also note the significant difference in energy between training and test sequences, which suggests limited generalization capacity. Still, as shown in Figure 4.8D, sequences generated at $T = 1$ are far from being simple copies of natural sequences.

10: In [58], the authors do not distinguish between training and test datasets, so we assume all sequences were used for training and that the relevant comparison is between artificial and training-sequence energies.

11: We further note that this model produces artificial sequences whose statistics are comparably close to the test sequences statistics as those obtained from the training set. Supporting figures are provided in Appendix C.

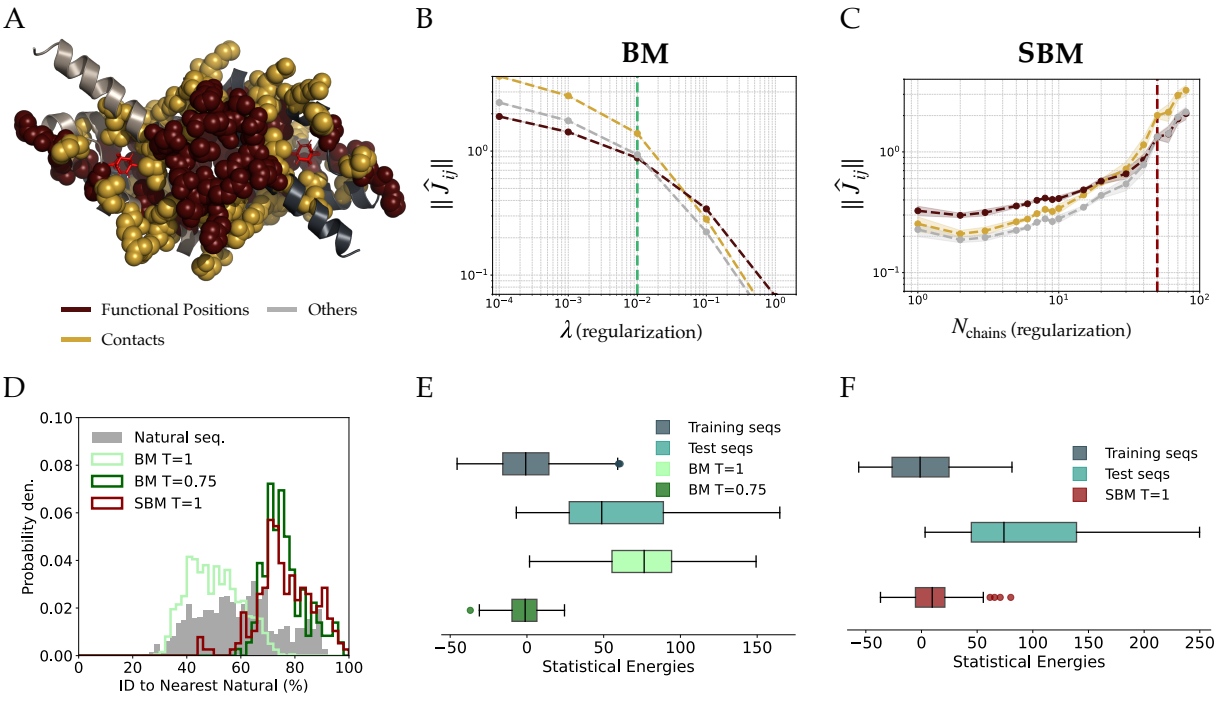

Figure 4.8: Comparison of BM and SBM methods on the Chorismate Mutase family.
A. Ribbon representation of *E. coli* CM domain structure (PDB: 1ECM). AroQ CMs are dimers with two symmetric active sites. The contacts and functional positions used in panel B & C are respectively highlighted with yellow and red spheres.
B. BM model: magnitude of inferred couplings as a function of L_2-regularization. At low regularization, contact-associated couplings dominate. As regularization increases, collective features become more prominent. The vertical green dotted line indicates the regularization level used in panel E.
C. SBM model: magnitude of inferred couplings as a function of the hyperparameter N_{chains} (note that larger N_{chains} corresponds to weaker regularization). A broad range of values yields accurate, unbiased estimates of interacting couplings. The vertical red dotted line marks the regularization level used in panel F.
D. Distribution of sequence identities to the nearest natural sequence (excluding self-matches) for different datasets. The grey histogram shows the distribution among natural sequences. Colored curves correspond to artificial sequences generated from BM at $T = 1$, $\lambda = 0.01$ (light green), BM at $T = 0.75$, $\lambda = 0.01$ (dark green), and SBM at $T = 1$, $N_{\text{chains}} = 50$ (red). Lower identity values indicate greater sequence novelty with respect to natural data.
E. Statistical energy distributions for training, test, random, and artificial sequences generated by the BM model at $T = 1$ and $T = 0.75$ (with $\lambda = 0.01$). Energies are calculated with the BM model at $T = 1$ and $\lambda = 0.01$. Artificial sequences at $T = 1$ show elevated energies; lowering T brings energies closer to those of natural sequences.
F. Same as panel E, but for an SBM model inferred with $N_{\text{chains}} = 50$, $m = 1$, and $T = 1$. Energies are calculated with the SBM model at $T = 1$ and $N_{\text{chains}} = 50$. Artificial sequences exhibit energy distributions comparable to those of the training set, without requiring temperature rescaling.

4.4 Protein design

We present here experimental results for the BM and SBM models introduced in the previous section. These experiments were conducted by Emily Hinds at the University of Chicago, following a protocol similar to the one described by Russ *et al.* [58], which we briefly summarized in Section 3.4.1.

4.4.1 Sampling at T=1: Can we gain in diversity?

In this section, we compare the generative capabilities of the two previously discussed models: BM ($\lambda = 0.01$, T=0.75) and SBM ($N_{\text{chains}} = 50$, T=1), referred to hereafter simply as BM and SBM.

Specifically, we have relative enrichment measurements for 890 natural sequences, 193 sequences generated using SBM, and 180 sequences generated using BM.

Figure 4.9A & B display these enrichment values along with the identity to the closest natural sequence for all tested sequences.

As previously mentioned, both models exhibit similar levels of novelty in the sense that they generate sequences with comparable average identity to the nearest natural sequence ($\sim 77\%$).

The distribution of relative enrichment is bimodal in each case and the enriched mode of natural sequences comprises 51.5% of the dataset.

For synthetic datasets, we obtain 26.7% of functional sequences with BM and 20.7% with SBM. Importantly, these functional sequences are not necessarily the closest to natural sequences, as their average identity to the nearest natural sequence remains around 77%. For instance, one functional sequence generated by SBM shares less than 50% identity with its closest natural counterpart.

These observations suggest that both BM and SBM capture sufficient information from the training data to sample a substantial number of functional sequences with an average similarity of 77% to the closest natural sequence.

But what about the diversity of these synthetic sequences? In Section 3.4.2, we highlighted the deleterious impact of lowering the temperature on the diversity of artificial datasets.

Figure 4.9C & D show the distribution of pairwise sequence identities for natural sequences, as well as those generated by BM and SBM. For BM at T=0.75, a second mode appears in the identity distribution, similar to what was observed in Russ *et al.*'s data.

The distribution for SBM, by contrast, suggests that there is no comparable loss in diversity because there is no temperature rescaling.

To further support this result, Figure 4.9E displays the percentage of sequences belonging to the most represented families in the natural dataset, alongside the proportion of functional sequences in each family[12].

When applying the same analysis to the two synthetic datasets, we find that 25.4% of the natural families are represented among the 180 sequences generated with BM T=0.75, compared to 35.8% for the SBM T=1 sequences[13].

12: In terms of diversity, it is worth noting that the average pairwise identity between natural sequences is 21%. On average, a given sequence has about 20 neighbors with more than 50% identity; this number drops to fewer than 10 when using a 60% identity threshold, and to fewer than 4 at 70%.

13: As in Section 3.4.2, each artificial sequence is assigned to the family of its nearest natural counterpart.

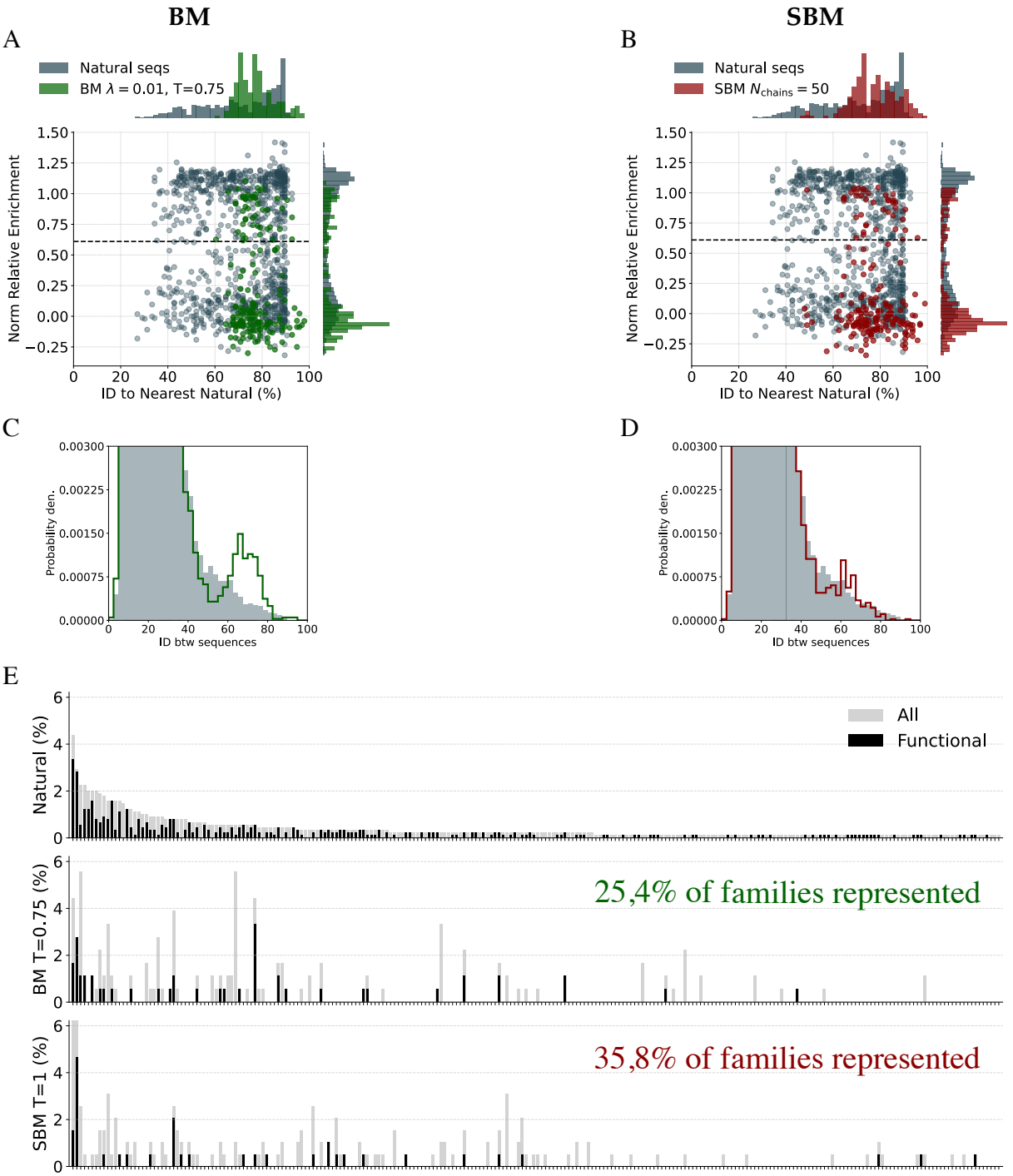

Figure 4.9: BM VS SBM generative power. (Panels A & B are made from data provided by Emily Hinds.)
Panels in the left column correspond to a BM model inferred with $\lambda = 0.01$, with sequences sampled at T=0.75, while panels in the right column correspond to an SBM model inferred with $N_{\text{chains}} = 50$ and sequences sampled at T=1.
A & B. Scatter plots showing the relative enrichment measured experimentally as a function of identity to the nearest natural sequence, for natural sequences (blue), BM-generated sequences (green, panel A), and SBM-generated sequences (red, panel B). The fraction of functional sequences for natural, BM, and SBM are respectively: 51.5%, 26.7%, and 20.7%.
C & D. Histograms of pairwise sequence identities among natural sequences (blue), BM-generated sequences (green, panel C), and SBM-generated sequences (green, panel D) for $N_{\text{chains}} = 50$.
E. The top bar plot shows the proportion of natural sequences belonging to the families represented in the natural alignment. For comparison, analogous statistics are shown for the 193 BM (middle) and 180 SBM (bottom) artificial sequences tested. Each artificial sequence is assigned to the family of its nearest natural counterpart. Percentages computed over 10,000 generated sequences are 58.8% for BM T=0.75 and 82.1% for SBM T=1.

These percentages remain relatively low, as they are computed over the experimentally tested sequences, which are fewer than the total number of natural families. Thus, when generating 10,000 sequences for each model, we find 58.8% family coverage for BM T=0.75 and 82.1% for SBM T=1 (for reference, this value is 93.8% for BM at T=1, which is known to lack generative capacity, according to the results of Russ *et al.*)[14].

14: Histograms are provided in Appendix C.

These findings suggest that the SBM model, due to its capacity to generate functional sequences at T=1, is better able to reproduce the diversity observed in natural datasets.

Another question is whether we can enhance novelty by increasing the regularization of the SBM to generate sequences that are more dissimilar from natural proteins. This is the focus of the next section.

4.4.2 Lowering the N_{chains}: Can we gain in novelty?

Building on the results obtained with the SBM at $N_{\text{chains}} = 50$, we now investigate whether we can enhance novelty, i.e., generate sequences that are more distant from natural ones.

Indeed, Figure 4.8D shows that the average identity to the nearest natural sequence is 77% for artificial sequences, compared to 60% for the natural dataset (based on the weighted statistics used to train the model).

For SBM with $N_{\text{chains}} = 50$, this identity distribution ranges from 44% to 99%, whereas for natural proteins it ranges from 25% to 92%. Additionally, as shown in Figure 4.8F, the average energy of the test sequences is significantly higher than that of the training sequences.

Can we further increase regularization, i.e., reduce the N_{chains} parameter, to generate sequences that are, on average, more divergent from natural ones?

To answer this question, we trained models with $N_{\text{chains}} = 20$ and $N_{\text{chains}} = 30$. Figure 4.10 presents model statistics and experimental results for 196 artificial sequences generated by the $N_{\text{chains}} = 30$ model[15].

15: Note that these models were inferred using a different parameter initialization (starting from a profile model instead of zero, as was the case for the $N_{\text{chains}} = 50$ model), and without the averaging procedure. In terms of couplings, we were not able to observe any clear differences between these inference settings, but new experiments are currently underway to address this question.

In Figure 4.10F, we observe that the energy distributions for the test and training sets do not completely overlap. However, the increased regularization brings the two distributions closer, indicating improved generalization capacity.

While the overlap between the artificial and training energy distributions is not as strong as in the SBM $N_{\text{chains}} = 50$ case, 99% of the artificial sequences still have energies below the maximum statistical energy observed in the training set.

In terms of novelty, Figure 4.10G shows that the identity-to-nearest-natural distribution now ranges from 32% to 80%, with an average of 56%. Importantly, sequences are sampled at T=1, preserving dataset diversity, as shown in Figure 4.10E.

Of the 196 artificial sequences tested, 6 exhibit relative enrichment levels indicative of functionality, representing 3% of the dataset. These functional sequences are among the closest to natural ones, with 69–73% identity to the nearest natural sequence.

The generative capacity of this model is therefore limited to a small fraction of functional sequences, which are not substantially more distant from natural sequences than those obtained with the $N_{\text{chains}} = 50$ model.

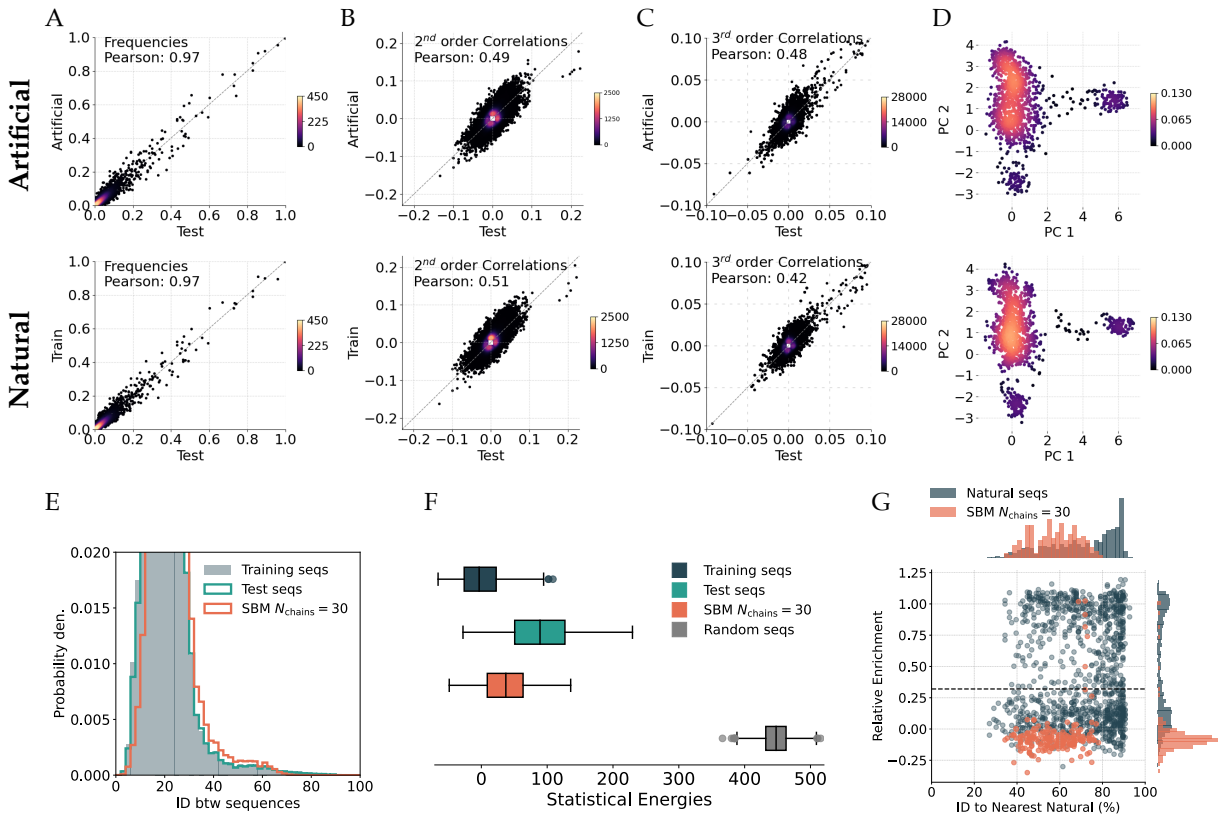

Figure 4.10: SBM Statistics and generative power. (Panel G is made from data provided by Emily H.)
A. Comparison of frequencies computed on the test and artificial sequences. The bottom panel shows the same comparison between the training and test sequences. The Pearson correlation coefficient is also provided. The color scale indicates point density.
B. Same as panel A, but for 2nd-order correlations.
C. Same as panel A, but for 3rd-order correlations.
D. Artificial sequences projected onto the first two principal components computed from the natural alignment. The bottom pannel shows the same projection for natural sequences for comparison.
E. Histograms of pairwise identity for training, test, and artificial sequences generated with SBM $N_{\text{chains}} = 30$.
F. Boxplots of statistical energy distributions for training, test, artificial, and random sequences.
G. Scatter plot showing relative enrichment as experimentally measured.

Nonetheless, this result is instructive as it prompts us to reconsider how generative models are commonly evaluated.

When comparing frequency statistics, second- and third-order correlations between the artificial, training, and test datasets, we observe that the similarity between artificial and test sets mirrors that between the training and test sets[16].

Moreover, principal component analysis shows that artificial sequences exhibit a distribution similar to natural sequences in the space of the first two components[17].

These statistical analyses are standard in the literature on generative models for protein sequences [43, 66, 80, 88, 96, 97], and are often regarded as strong in silico indicators of a model's generative capacity.

However, as already shown by Russ *et al.*, matching these statistics does not guarantee generative capacity, as illustrated, for instance, by the BM at $T = 1$. Similarly, models that reproduce the energy distribution of the training set, such as null or profile models, cannot automatically be considered generative.

Here, we extend this observation by showing that even a model that captures standard statistical features and produces sequences with training-

16: For these models, the test set was designed to exclude any sequence with more than 80% identity to a training sequence.

17: Note, however, that the variance explained by these two components is less than 8%.

like energies may still fail to generate a significant fraction of functional sequences.[18]

18: This conclusion also holds for the $N_{\text{chains}} = 20$ model; see Appendix C for details.

4.5 Discussion

4.5.1 Summary

This chapter introduced the Stochastic Boltzmann Machine (SBM), an inference method based on the L-BFGS algorithm, with an implicit regularization scheme based on the hyperparameters: N_{iter}, N_{chains}, and m.

Its performance was first assessed on a toy model designed to reflect interactions at multiple scales within sequence data. On this model, the use of L-BFGS led to distinct optimization trajectories in parameter space, bringing the inferred couplings closer to an unbiased estimate of the true interactions.

Beyond this, we explored how m and N_{chains} can be used to modulate the strength of regularization. A systematic study of inferred couplings across different values of N_{chains} revealed that higher values act similarly to weaker L_2 penalties in standard Boltzmann Machines. Interestingly, lowering N_{chains} allowed us to reach a regime where inference bias is reduced, while preserving proximity to the ground-truth parameters.

Motivated by these results, the method was then applied to real protein data, focusing on the chorismate mutase family. Several models were tested on this dataset, with both statistical and experimental evaluations.

At temperature $T = 1$, we showed that the SBM was able to generate artificial sequences with energies comparable to those observed during training, an outcome not achieved by the standard BM except under very weak regularization. When comparing models with similar levels of novelty, BM ($T = 0.75$, $\lambda = 0.01$) and SBM ($T = 1$, $N_{\text{chains}} = 50$) generated comparable fractions of functional sequences. Yet, because the SBM does not require low-temperature sampling, it is able to better reproduce the diversity observed in natural sequences.

We then explored the effect of increased regularization (i.e., smaller values of N_{chains}) with the goal of generating sequences more distant from natural ones. With $N_{\text{chains}} = 20$ and 30, the models generated very few functional sequences, all of which were among the closest to natural ones.

These results first underscore the limitations of the SBM approach, but also raise a broader question: to what extent is it possible to generate functional sequences that are substantially different from natural ones, given the available data and the modeling framework used?

Moreover, the failure of these models to produce functional sequences stands in contrast to classical in silico analyses, which had suggested a statistical resemblance between artificial and natural sequences. This discrepancy highlights the limitations of such analyses in reliably evaluating the generative capacity of a model.

4.5.2 About Generative Model Evaluation

Since the early 1990s, the literature on generative models for protein sequences has grown considerably,[19] with a wide variety of models being proposed: VAEs, autoregressive models, generative adversarial networks, diffusion models, transformers, energy-based models, and others.

19: On PubMed, the number of hits for the keywords "generative model protein" increased from fewer than 1,000 in the early 1990s to over 17,000 in 2024.

However, given the high cost and technical expertise required for experimental validation, only a limited number of models are actually tested in the lab.

Many of the proposed models are typically evaluated in silico across multiple protein families to demonstrate their ability to generate sequences that are statistically indistinguishable from natural ones.

Evaluations often rely on comparing statistics between natural and synthetic sequences, leveraging more complex metrics derived from other models (e.g., AlphaFold structure prediction confidence scores), or assessing the model's ability to predict contacts or mutational effects [43, 66, 80, 88, 96, 97].

While these in silico evaluations are far from uninformative, they deserve critical examination.

For instance, should a Potts model designed to generate protein sequences necessarily perform optimally in contact prediction? Especially if improved contact prediction comes at the expense of capturing other types of interactions?

Should we rely on AlphaFold scores to distinguish between good and bad generative models?

Consider AlphaFold's confidence score (pLDDT): when confronted with experimental data, it shows a large number of true positives and true negatives, but also a non-negligible number of false positives[20]. For example, artificial sequences generated with SBM $N_{\text{chains}} = 50$ and SBM $N_{\text{chains}} = 30$ achieve average pLDDT scores of 93 and 92, respectively[21]. In this case, one cannot claim that AlphaFold could have predicted the non-functionality of SBM $N_{\text{chains}} = 30$ sequences.

20: It has, for example, been shown that AlphaFold cannot be used to predict the effect of single mutations on protein stability and function [117].

21: These scores were computed over 50 to 150 sequences for both artificial and natural datasets, and are provided in Appendix C. Additional statistics will be required to draw more robust conclusions.

But is that even desirable? Might it not be just as interesting to study models that AlphaFold fails to validate? Given the high cost of experiments, it may seem unwise to test a model that generates sequences for which AlphaFold cannot confidently predict a structure. Yet this is precisely where "true" novelty may lie.

Several studies have addressed the problem of evaluating generative models [92], often in the context of images [118], but also more specifically for protein sequences. As mentioned in Section 2.3.5, one proposed approach involves the use of synthetic data, commonly generated from a model (often a Boltzmann Machine) trained on real data [102, 103].

However, this approach implicitly assumes that the generative model used to produce synthetic data has accurately captured the relevant structure of the original data and is free from significant biases. Otherwise, evaluating new models on such synthetic data may be misleading. To exaggerate: one could imagine evaluating a model using data generated by an independent model. More subtly, consider a Boltzmann Machine that captures local interactions very well but fails to represent collective ones. If such collective interactions are actually important, then we risk overlooking a critical aspect in model assessment.

Rather than using such synthetic data, we opt for a toy model not trained on real sequences but specifically designed to reflect what we believe to be important features of protein sequences: namely, a network of multiscale interactions, which in real data plays a fundamental role in structure, function, and evolvability.

This choice is supported by numerous studies showing that, beyond residue-residue contacts corresponding to 3D structural proximity, groups of co-evolving residues can be crucial for protein function. Some of these groups—referred to as sectors—have been experimentally characterized in diverse systems, such as serine proteases [19], dihydrofolate reductases [36], and rhomboid proteases [41].

Of course, building a toy model carries the risk of misidentifying what is essential and what is not. Yet this constraint forces us to think critically about the features we consider fundamental. In that sense, such models can serve as useful tools to isolate and test specific hypotheses about the principles underlying protein properties.

This naturally raises a broader question: to what extent can generative models help us not only generate sequences, but also deepen our understanding of the mechanisms that govern protein function?

4.5.3 About the utility of these generative models

There are multiple motivations for developing generative models of protein sequences.

One practical goal is to design protein sequences, distinct from those found in nature, that perform a specific biological function. This might involve improving properties such as thermostability or catalytic efficiency of a given protein, or modifying a protein's function—for example, modifying an enzyme to act on a new substrate [119, 120].

Beyond this design-oriented goal, generative models can also offer a framework for understanding. By analyzing why a model succeeds, or fails, to produce functional sequences, we can gain insight into the fundamental features that govern protein properties. In this context, prediction becomes a means rather than an end and the objective is to understand what features a model must capture in order to generate functional sequences as diverse as natural proteins, without copying them.

In this work, we focused on a class of models that aim to reproduce the empirical first- and second-order statistics of natural sequences. Experimental data, such as that from the chorismate mutase family [58], show that first-order constraints alone are not sufficient to produce functional sequences.

Incorporating second-order statistics, as in the Boltzmann Machine (BM), requires low-temperature sampling to generate functional sequences. However, we have shown that such sampling leads to a substantial reduction in sequence diversity. This suggests that the Boltzmann Machine (BM) model, in its current form, may be inadequate.

This naturally raises the question of whether the model class itself should be reconsidered. In this work, we chose to retain the Potts model framework and instead focused on the inference procedure, hypothesizing that improved inference could better capture the relevant structure of the data.

One of the key findings of this work is that a Potts model inferred using the SBM method more faithfully reproduces the diversity observed in natural data, thanks to its ability to generate functional sequences at $T = 1$.

But what does this result teach us about proteins? What lies at the heart of this SBM model's success? Does it stem from the correction of biases between local and collective interactions, as observed in the toy model? At this stage, we do not yet have sufficient evidence to answer these questions with confidence.

Nonetheless, these are precisely the kinds of questions we must address if we aim to move from predictive performance toward deeper understanding of proteins.

4.5.4 Future Directions

Despite the promising results of the SBM approach, we still lack a clear theoretical understanding of how it works. Building such a foundation is an important next step. One way is to explore simplified models where the inference dynamics can be worked out exactly, gradually adding the features that define the SBM. This is the direction we are currently taking with Milo Repossi, a student who has been working since January on Gaussian models to study how the structure in the data affects inference.

A natural follow-up question is whether the biases observed in BM models also appear in other generative frameworks, such as Restricted Boltzmann Machines, diffusion models, or transformers. To answer this, we would eventually need a way to identify such biases that doesn't depend on the model itself.

Interestingly, it seems that a mapping exists between the inverse Potts model and other architectures. For instance, it was recently shown that a single factored attention layer trained with masked language modeling is equivalent to solving the inverse Potts model using the pseudolikelihood approximation [121]. While such models differ from large off-the-shelf transformers pretrained on billions of sequences, they could provide a useful starting point for investigating undersampling biases in these architectures.

On the application side, it would be interesting to move beyond the chorismate mutase family and apply SBM to more complex systems. Serine proteases, discussed in Chapter 6, are a good candidate: they are longer ($L \sim 250$), show remarkable functional diversity and have already been widely studied through the SCA approach, which revealed three distinct sectors tied to specific biochemical properties.

Beyond the distribution of natural protein sequences

Protein Properties and Mutational Effects

5

This third part of the thesis explores how energy-based models can guide the prediction of compensatory mutations in protein sequences with the objective of: (1) investigating the determinants of specificity within the serine protease family, and (2) supporting the understanding of an allosteric mechanism in the dihydrofolate reductase family.

Chapter 5 outlines some important concepts linked to protein properties, functions and mutational effects. Chapter 6 presents the proposed approach and its application to serine proteases. Finally, Chapter 7 extends the method to an other context: compensating for a mutation within the allosteric network of *E. coli* dihydrofolate reductase.

5.1 Proteins, physical properties and function

The biological functions of proteins emerge from their chemical and physical properties, which are governed by the collective interactions of amino acid residues within their structure. Subsequent sections provide a concise summary of some experimental and computational techniques developed to analyze protein structure, dynamics and function under varying environmental conditions.

5.1.1 Physical properties

Among the physical properties of proteins that can be characterized, one of the most important is their conformation, i.e., the folded structure dictated by the amino acid sequence[1] . This structure can be determined experimentally through techniques like x-ray crystallography [12] that require generating a suitable protein crystal to study the diffraction pattern of x-rays passing through it, or nuclear magnetic resonance (NMR) [13] that can also be used to monitor changes in protein structure (for example during folding or binding to another molecule).

1: A protein that has been purified and denatured by a solvent often refolds spontaneously when the solvent is removed, indicating that the sequence contains all the information necessary to specify the structure [6].

These experimental structure data, combined with the vast amount of protein sequences now available [23], can be used to train models that accurately predict the structure of many proteins. In this context, the power of deep learning became especially clear after the first AlphaFold's [14] victory at the Critical Assessment of Structure Prediction (CASP) competition in 2018.

Experimental 3D structures can also be integrated into physical models. As we will see in Chapter 7, when applied to proteins, molecular dynamics simulations offer valuable insights into biological phenomena, such as conformational changes caused by mutations.

Another key property of proteins that has also been widely studied is the stability, which refers to the ability to maintain the functional structure under varying conditions. For example, the thermodynamic stability is commonly assessed with techniques such as Differential Scanning Calorimetry (DSC) [122], that can be used to quantify the temperature at which the protein denatures or unfolds.

Beyond simply maintaining their structural integrity, proteins must also interact with other molecules. The strength of binding between two molecules (A and B) can be quantified through the equilibrium constant of the reaction: $A + B \rightleftarrows AB$. This binding affinity of a protein for another molecule, called a ligand, is a critical property because such interactions govern the function of most—if not all—proteins, from antibodies that bind to viruses to enzymes that interact with ligands in order to catalyze chemical reactions.

5.1.2 Function and fitness

Natural evolution has selected proteins for their function, i.e. their role in living organisms[2]. In fact, variations in a protein's function can greatly influence the organism's capacity for survival, growth, and reproduction in a given environment, collectively referred to as fitness. It is important to note that even a single amino acid mutation can result in a complete loss of the protein function.

2: In addition to function, evolvability is also often considered a selectable trait.

There is no straightforward approach to determine a protein's role in a given organism when it is unknown. Nevertheless, comparing the sequence of a newly identified protein to existing databases of known proteins can often provide significant insights regarding its role.

In the following chapters, we will concentrate on two protein families with established roles in living organisms, supported by experimental data concerning their structure and function. These proteins are part of the large functional category of enzymes. While enzymes have been discussed previously, we will briefly describe the experimental methods employed to measure these proteins' activities.

5.1.3 Kinetics of enzymes

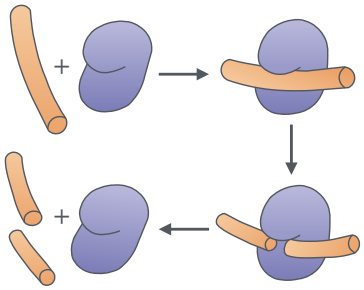

Figure 5.1: Schematic representation of an enzyme catalyzing a cleavage reaction. The enzyme first binds to the substrate to form the enzyme-substrate complex, then catalyzes the cleavage of a covalent bond, and finally releases the products.

An enzyme accelerates a reaction without undergoing any change itself, thus functioning as a catalyst. Enzymes can be classified into several categories, each highly specialized and effective in catalyzing a particular type of reaction. For instance, hydrolases are enzymes that catalyze the cleavage of chemical bonds using water.

A widely used and effective model is the Michaelis-Menten model [123], which provides a simple yet powerful framework to describe enzyme kinetics. According to this model, an enzyme E must first bind to a ligand called substrate S to convert it into a modified product P:

$$\mathrm{E} + \mathrm{S} \underset{k_-}{\overset{k_+}{\rightleftharpoons}} \mathrm{ES} \xrightarrow{k_{\mathrm{cat}}} \mathrm{E} + \mathrm{P}. \tag{5.1}$$

When all enzymes are occupied by a substrate, the reaction rate only depends on how quickly the enzyme can process the substrate. This maximum rate divided by the enzyme concentration is referred to as the turnover number, denoted as k_{cat}.

Another frequently used parameter to characterize enzymes is the Michaelis constant K_{m}, which represents the substrate concentration at which the reaction rate reaches half of its maximum value (Figure 5.2). Thus, a low K_{m} means that the enzyme achieves its maximum catalytic capacity at low substrate concentrations, suggesting a strong affinity for its substrate.

The ratio $\frac{k_{cat}}{K_m}$ is widely used to assess catalytic efficiency, either by comparing the same enzyme acting on different substrates, or by comparing different enzymes with their respective substrates.

To obtain an accurate measurement of this quantity, it is necessary to conduct multiple experiments at varying substrate concentrations in order to obtain kinetic curves. As we will see in Chapter 6, such curves can be obtained through fluorescence measurements using a substrate that becomes fluorescent when processed by the enzyme.

In the following, we will focus on two families of such proteins called enzymes for which experimental methods exist to measure their efficiency. Since these proteins have already been extensively studied as we will see in chapter 6 and 7, they serve as suitable model systems to study important properties of protein functions.

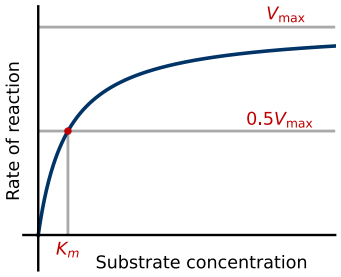

Figure 5.2: The reaction rate of an enzyme increases with the substrate concentration until a maximum value V_{max} corresponding to the situation where all enzymes are occupied with a substrate. The concentration of substrate at which the reaction rate reaches half of its maximum value is denoted as K_m.

5.2 Mutational Effects

Assessing how mutations can impact a protein's stability, function or the fitness of an organism provides crucial insights into the sequence-function relationship. Thus, extensive research has been conducted to investigate these mutational effects through both experimental approaches and computational tools.

5.2.1 Experimental Measurement

We have already mentioned that organisms (e.g., bacteria) can be modified to express mutant or artificial proteins while the gene coding for the native version is knocked out. Suppose we want to test all possible single mutations in a 300-amino-acid protein, which amounts to 6,000 mutants. Without high-throughput methods, this would require a considerable experimental effort. However, since the 2010s, Deep Mutational Scanning (DMS) approaches combined with next-generation sequencing have enabled systematic evaluation of mutations' impacts on protein stability, activity, binding affinity, and organism fitness [124].

These experiments typically begin with the generation of a comprehensive library of mutant variants[3]. This library is then exposed to a selection or screening process specifically designed to probe the property or function of interest. The effect of each mutation is finally assessed using an enrichment metric, which compares the abundance of each variant before and after selection. To evaluate whether a mutation is beneficial or deleterious, it is common practice to normalize the enrichment of each mutant by that of the non-mutated, or wild-type, protein, and finally take the log of this quantity[4].

This systematic evaluation of individual mutations can reveal unexpected insights, such as mutations critical for function even if distant from the active site. This phenomenon, known as allostery, will be discussed further in Chapter 7.

If we assume that all residues in a protein are independent, a single DMS experiment would be sufficient to predict the impact of mutations on all sequences within the same family. However, empirical evidence suggests that this is not the case. Experiments measuring the effects of single and double mutations on a protein have, for example, demonstrated

3: These variants are typically associated with barcode sequences, that are short DNA fragments uniquely identifying each variant. This allows for the rapid tracking and quantification of all mutants in parallel through sequencing techniques.

4: This quantity will be denoted as log(r.e.) for logarithm of relative enrichment. When > 0 it thus means that the mutation is beneficial with respect to the Wild Type.

that certain mutations can have different effects when combined to other mutations, a phenomenon known as epistasis [125, 126].

5.2.2 Epistasis

First described in genetics (1907) [127], epistasis refers to the genetic background dependency of mutation effects [128]. In other words, combinations of mutations can lead to effects that are not predictable from single mutations alone[5]. These interactions typically occur at various levels, from pairwise interactions to more complex higher-order combinations of mutations. Depending on the direction and the magnitude of the resulting effects, epistatic interactions are usually classified into different subtypes. In the following, we will, for example, refer to positive epistasis, where the combined effect of mutations is more beneficial than expected from their individual contributions.

5: Defining epistasis as the non-additivity of mutations first requires a clear baseline for what constitutes independent mutation effects. Except when considering a thermodynamic state variable, establishing such a baseline is far from straightforward [129].

Extensive research has sought to uncover the mechanisms underlying epistasis and its impact on protein evolution [130, 131]. To address these questions, high-throughput screening methods have enabled the quantification of epistatic interactions in the local sequence neighborhood of some proteins [125, 132, 133].

In 2014, Olson *et al.* [125] generated a log-enrichment map of single and double mutants in a bacterial protein domain that binds antibodies. Their experiments revealed that most deleterious mutations have one or more interacting mutations that render them either beneficial or neutral. One year later, Podgornaia *et al.* [133] conducted a high-throughput screen targeting four key residues of the *E. coli* protein kinase PhoQ and found that 94% of the functional variants identified would have been predicted to be nonfunctional based on the effects of single mutations alone (Figure 5.3).

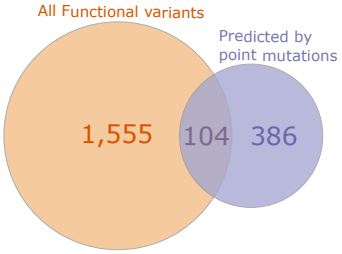

Figure 5.3: Epistasis in *E. coli* PhoQ. (Adapted from [133])
Venn diagram illustrating the overlap between functional PhoQ variants identified experimentally and those predicted from single mutants, assuming independence between positions.

These studies suggest that epistasis is abundant in the local sequence neighborhood of a protein. In particular, positive epistasis appears to be a key factor in protein evolvability, as it can open new mutational pathways. For instance, in the TEM-1 β-lactamase antibiotic resistance gene, it has been shown that a stabilizing mutation can mitigate the otherwise deleterious effects of mutations that increase the protein's activity against novel antibiotics [74, 134]. These results suggest that the local sequence landscape of a protein may exhibit a great functional diversity, and epistasis may facilitate transitions between these distinct functional regions within sequence space.

In Chapter 6, we will focus on the serine protease family. These enzymes catalyze cleavage reactions and are each highly active on different substrates, thus displaying a range of specificities. For this family, DMS datasets for combinations of mutations are not available. However, in 1992, Hedstrom *et al.* demonstrated that it was possible to significantly change the specificity of one such enzyme with only a few mutations. Since then, other attempts have been made; some have been unsuccessful, and most have resulted in only partial conversions.

Taken together, all these studies highlight the complexity of epistatic interactions and the potential that a deeper understanding of these effects could offer.

5.2.3 Predict epistatic interactions

Functionally or structurally important couplings between amino acids are expected to drive their coevolution, leaving detectable signatures of covariation in multiple sequence alignments that reflect the evolutionary history of a protein family. Consequently, data-driven models trained on such datasets can be used to predict epistatic interactions.

A wide range of computational approaches based on sequence data have demonstrated their ability to predict residue-residue contacts within a protein's tertiary structure, as well as to estimate the effects of single or multiple mutations. These include supervised methods relying on single-site conservation [135], unsupervised statistical models incorporating pairwise interactions such as Direct Coupling Analysis (DCA) [75, 77], approaches that explicitly model the evolutionary history of protein families [136], and deep learning-based models [137]. [6]

6: All of these methods have been benchmarked against Deep Mutational Scanning (DMS) datasets. However, their performance varies considerably across different protein families. The best-performing method is not always the same, and for some families, such as viruses, the independent model can even outperform "epistatic" models that explicitly account for amino acid interactions [137–139].

In the following chapters, we focus on two systems in which the goal is to identify mutations that are epistatically coupled to a known deleterious mutation. Specifically, the aim is to propose compensatory mutations that can rescue the functional impact of such a deleterious change. This strategy will serve two different purposes: (1) further explore the determinants of specificity within the serine protease family, (2) support the understanding of an allosteric mechanism in *E. coli* dihydrofolate reductase.

To this end, we will employ the Boltzmann Machine (BM) framework introduced earlier in this thesis. This model defines a probability distribution over sequence space, constructed using the Maximum Entropy Principle. Its parameters are fitted by enforcing constraints on empirical statistics through an optimization procedure.

The probabilities assigned by the model inherently depend on the data used for training—particularly the number and diversity of available sequences—as well as the type of statistical features included (here, single-site and pairwise amino acid frequencies). Our goal is to use these probabilistic predictions not as an end in themselves, but as tools to gain deeper insight into the biological systems under study.

Specificity Conversion in Serine Proteases

6

The following chapter is based on a collaborative effort with the team led by Clément Nizak at the Jean Perrin laboratory.

Amaury Paveyranne, a PhD student co-supervised by Clément Nizak and Olivier Rivoire, works on a specific family of enzymes through computional and experimental approaches. This family considered as a model system is notably known for its functional diversity. Significant efforts have been made to understand the mechanisms driving this diversity, with a particular emphasis on the ability to switch protein functions through a limited number of mutations. The objective of this work is thus to explore how the BM model can be used to guide such a conversion.

My contribution to this work involved the development of the prediction methodology. The experimental asssays to measure the activity of the different mutants were conducted by Amaury Paveyranne, Timothé Lucas and Shoichi Yip.

As this is an ongoing project, the results presented in this chapter should be considered preliminary.

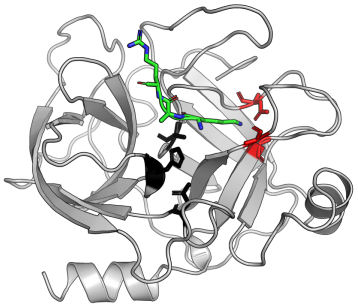

Figure 6.1: rat trypsin structure. Ribbon representation of rat trypsin structure (PDB: 3TGI) with an inhibitor protein highlighted in green sticks, the catalytic triad in black and the binding pocket in red (visualized with Pymol open source).

6.1 Serine Proteases

Serine proteases represent a large family of functionally diverse enzymes that play essential roles in a wide array of physiological processes, including digestion, immune response and blood coagulation [21, 140]. Despite their involvement in distinct biological pathways, all serine proteases catalyze the hydrolysis of peptide bonds through a similar mechanism centered around a serine residue within their active site that gave the family its name [141]. In the following sections, we will focus on the specific S1A subfamily within Protease Clan PA, which represents the largest and arguably one of the most studied enzyme groups.[1]

Two groups of residues have been identified as essential for the activity of these proteases:

- The catalytic triad composed of serine (Ser195)[2], histidine (His57), and aspartate (Asp102) residues.
- The binding pocket shaped as a cavity by three residues interacting with the substrate and connected to the active site.

Although serine proteases share a common fold and catalytic mechanism, their specificity profiles can vary significantly depending on the position of the substrate. The residue located immediately upstream of the cleavage site is commonly referred to as P1. Trypsin and chymotrypsin are two examples of digestive serine proteases with different specificity profiles. In particular, trypsins prefer Arg/Lys side chains at the P1 position of substrate but chymotrypsins prefer Phe/Tyr/Trp residues.[3,4] The structure of rat trypsin is shown in Figure 6.1, highlighting the catalytic triad and the binding pocket.

The specificity of these proteases has long been attributed to the geometry of the binding pocket and the electrostatic interactions that occur within it.

1: The S1 family belongs to the PA clan and is further divided into subfamilies S1A and S1B [21].

2: The notation Ser195 indicates the presence of a serine residue at position 195. In the following, when referring to specific residue positions in a serine protease, we will adopt the residue numbering from the PDB structure 3TGI.

3: For highly specific family members such as trypsin, the catalytic efficiency ratio (k_{cat}/K_m) between a lysine and a phenylalanine substrate can reach values on the order of 10^5 [142].

4: Serine proteases are not exclusively sensitive to the amino acid at the P1 position; rather, they recognize a short motif within the substrate. While residues downstream of P1 also contribute to specificity, their influence is more limited and will be neglected in the remainder of this chapter.

Table 6.1: Table summarizing the various attempts to convert specificity in the S1A family. The table includes the activity achieved as a percentage of the target enzyme's amidase activity for k_{cat/K_m} measurements † or absorbance measurements ‡. Those related to esterase activity instead of amidase activity are indicated by ⋆.

Paper	Year	Conversion	# of mutations	Result
Hedstrom *et al.* [143]	1994	Rat tryp. to chymo.	16	2 – 15% †
Caputo *et al.* [144]	1994	Granz. B to chymo.	1	30 – 40% ‡⋆
Venekei *et al.* [145]	1996	Bovin chymo. to Tryp.	15	10^{-4} – 10^{-3}% †
Hung *et al.* [146]	1998	Rat tryp. to elastase	15	2% †⋆
Caputo *et al.* [147]	1999	Granz. B to tryp.	1	nd†
Jelinek *et al.* [148]	2004	Rat chymo. B to tryp.	2	0.01% †

In fact, the binding pocket of trypsin composed of Gly216, Gly226, and Asp189 which is negatively charged is, for example, more efficient at cleaving after amino-acids with positively charged side chains. Chymotrypsin, for its part, carries a Ser189 at the bottom of its binding pocket and can accommodate aromatic side chains.

This line of reasoning—linking specificity mainly to the geometry and chemical environment of the binding pocket—is commonly referred to as the structural approach.

However, attempts at specificity conversion in a few mutational steps have suggested that this approach alone is insufficient to explain the specificity of these enzymes, and the underlying mechanisms are not yet fully understood.

6.1.1 Attemps at specificity conversion

Since the 1980s, several studies have focused on the rules that encode specificity in proteases. Testing the structural approach hypothesis, which suggests the binding pocket controls specificity, led to attempts at converting trypsin into chymotrypsin. Since the binding pocket of these two proteases differs only at residue 189, it was expected that the D189S trypsin mutant would cleave chymotrypsin-type substrates.The verdict came at the end of the 1980s, when Graf *et al.* measured the catalytic activities of the rat trypsin D189S mutant, which were far below those of the natural homologs [149].

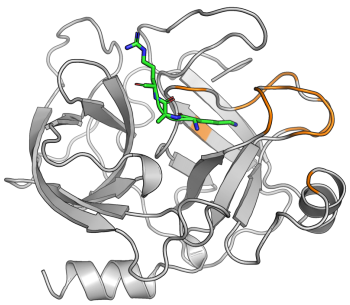

Figure 6.2: Hedstrom swap. Ribbon representation of rat trypsin structure (PDB: 3TGI) with the hedstrom's swap residues highlighted in orange.

In the early 1990s, Hedstrom *et al.* proposed complementing the structural approach with an evolutionary strategy by comparing trypsin and chymotrypsin sequences from various organisms [142]. This approach led to the first partial but successful conversion in rat trypsin, which required 15 mutations, mostly located on two loops that are not directly in contact with the substrate (see Figure 6.2). In 1994, they identified an additional mutation, which resulted in a protease exhibiting 2-15% of chymotrypsin activity [143]. This mutant is what we will refer to later when discussing Hedstrom's swap.

The same strategy was attempted for the reverse conversion without success [145], and the trypsin to elastase[5] switch also yielded modest results [146]. More recently, the work of Jelinek *et al.* revealed that the S189D mutation combined with the A226G mutation was sufficient to give rat chymotrypsin B 0.01% of a trypsin-like activity [148].

We can also note additional conversion attempts targeting other members of the S1A family, such as granzyme and thrombin [144, 147, 150, 151]. A

5: Elastase exhibits a strong preference for cleaving peptide bonds adjacent to small, non-polar amino acids, particularly Ala (A) and Gly (G).

non-exhaustive list of conversions explored in the literature is provided in Table 6.1.

These efforts to convert enzyme specificity have exposed the limits of purely structural models and highlighted the role of long-range, evolutionarily encoded interactions. Statistical coupling analysis (SCA) is one such approach developped to capture coevolutionary signals between residues within a protein family.

6.1.2 Specificity sector

When SCA was developed and applied to the S1A family, it revealed three quasi-independent groups of residues with distinct functional roles [19, 29]. One of these groups, referred to as the red sector, was identified as the main determinant of specificity in serine proteases of this family. Remarkably, most residues in this sector belong to, or are located near, the Hedstrom swap, as shown in Figure 6.3.

As already mentioned, the past three decades have seen a substantial increase in the amount of protein data available in public databases, making it possible to construct alignments with far more sequences than was previously feasible. A new MSA, built by Shoichi Yip as part of his Master's thesis, allowed us to revisit the original SCA analysis using an alignment with an effective number of sequences ten times larger than the original dataset[6]. As shown in Figure 6.3, the red sector obtained by applying SCA on this new dataset aligns with the results obtained from the Halabi *et al.* MSA.[7] This supports the use of this larger MSA for further evolutionary analysis.

While SCA has helped identify a set of residues involved in specificity—consistent with the earlier work of Hedstrom *et al.* [143] —, the conversion problem remains partially solved. A switch from chymotrypsin to trypsin has, for example, not yet been achieved, and most of the successful attempts at conversion remain partial.

To further explore the determinants of specificity within this protein family, we propose to build on the previously introduced energy-based model. Although residue 189 is recognized as a major determinant of specificity, it is not sufficient on its own to account for the full mechanism. We therefore initiate our analysis by introducing a mutation at this critical position and aim to predict compensatory mutations that could collectively shift substrate specificity.

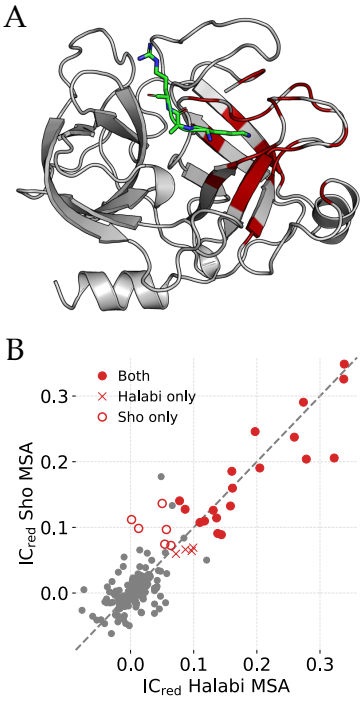

Figure 6.3: Robustness of the specificity sector to MSA choice.
A. Ribbon representation of rat trypsin (PDB: 3TGI) showing the specificity sector (red) identified by Statistical Coupling Analysis (SCA), using the larger MSA constructed by Shoichi Yip.
B. Comparison of residue contributions to the red (specificity) sector obtained from the two MSAs: the original one from Halabi *et al.* and the larger alignment built by Shoichi Yip. Red filled circles indicate residues identified in both MSAs; red crosses and open circles represent those specific to Halabi's or Shoichi's MSA, respectively. These results were obtained using the COCOATree toolbox [2].

6: The MSA from Halabi *et al.* has a size 1470 while the new MSA made by Shoichi Y. has a size 93695. Regarding the effective number of sequences it is typically 16806 against 799.

7: For a more comprehensive comparison of the SCA results, see Appendix D.

6.2 Difficulty of the task

We have just reviewed how the question of specificity in serine proteases has been addressed in the literature. Successful specificity conversions are rare, which suggests that the task is inherently difficult.

A first reason for this lies in the sheer size of the mutational space that becomes accessible as one moves away from a reference sequence. While there are only $20 \times L \sim 10^3$ single mutants, the number of n-point mutants is given by $\binom{L}{n}20^n$, yielding approximately 10^6 double mutants and 10^9 triple mutants.

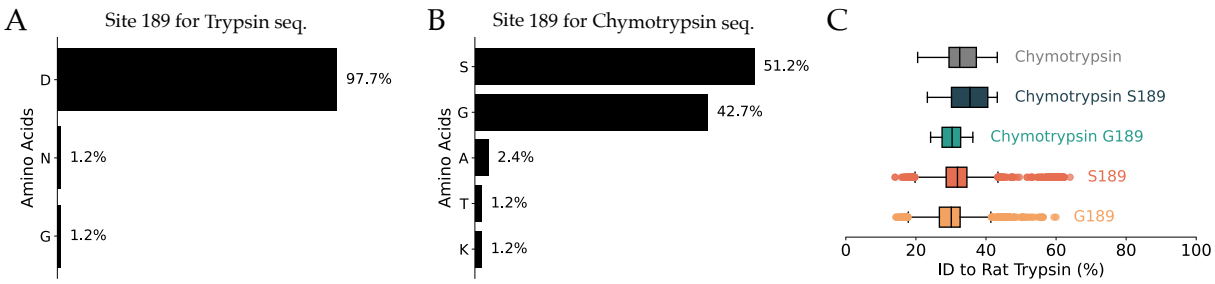

Figure 6.4: Statistics at site 189 in Serine Proteases.
A. Amino acid distribution at site 189 for sequences annotated as trypsin. The vast majority of trypsin sequences contain an aspartic acid (D) at site 189.
B. Amino acid distribution at site 189 for sequences annotated as chymotrypsin. The distribution is nearly evenly split between serine (S) and glycine (G).
C. Box plots showing percentage identity to rat trypsin for: chymotrypsin sequences (grey), chymotrypsin sequences with S189 (blue), chymotrypsin sequences with G189 (green), all sequences with S189 (orange), and all sequences with G189 (light orange).

Clearly, it is not feasible to experimentally test all these variants. This is why structural information and sequence data are essential tools for investigating the mechanisms of specificity.

We have emphasized that residue 189 has been structurally identified as a key determinant of specificity—a finding that is also supported by sequence data.

In Figure 6.4A & B, we show the amino acid distribution at site 189 across 172 sequences annotated as trypsin and 82 sequences annotated as chymotrypsin. The difference is striking: almost all trypsins carry an aspartic acid (D) at position 189, while chymotrypsins predominantly contain either a serine (S) or glycine (G).

If one aims to confer chymotrypsin-like activity onto the rat trypsin sequence (which will serve as our reference in the following), a logical starting point would be to introduce either the D189S mutation (as performed by Hedstrom) or the D189G mutation.

We may also ask whether any annotated chymotrypsin sequences in the dataset are already relatively close to the rat trypsin sequence. As shown in Figure 6.4, the closest chymotrypsin shares 43% sequence identity with rat trypsin (which corresponds to 111 mutations over an alignment of 260 positions)[8].

8: Although not shown here, the same pattern holds when comparing all annotated trypsins to all annotated chymotrypsins—the highest identity observed between any trypsin and any chymotrypsin in the alignment is 45%.

Interestingly, chymotrypsins that carry a glycine at position 189 tend to be, on average, more distant from trypsins than those with a serine at that position. When we remove the restriction to annotated chymotrypsins, the closest sequence to rat trypsin with either S or G at position 189 still shows only 64% identity (which corresponds to over 90 mutations across 260 positions).

However, the vast majority of sequences in the dataset lack any functional annotation. The closest annotated sequence to rat trypsin with a G189 is a kallikrein (which shares a specificity similar to that of trypsin) and the closest with a S189 is a chymotrypsin.

The limited sequence similarity between trypsins and chymotrypsins underscores the difficulty of rationally designing specificity-switch with only a few mutations. We now turn to a statistical modeling framework based on Boltzmann Machines, starting by examining the statistical energy scores provided by such models.

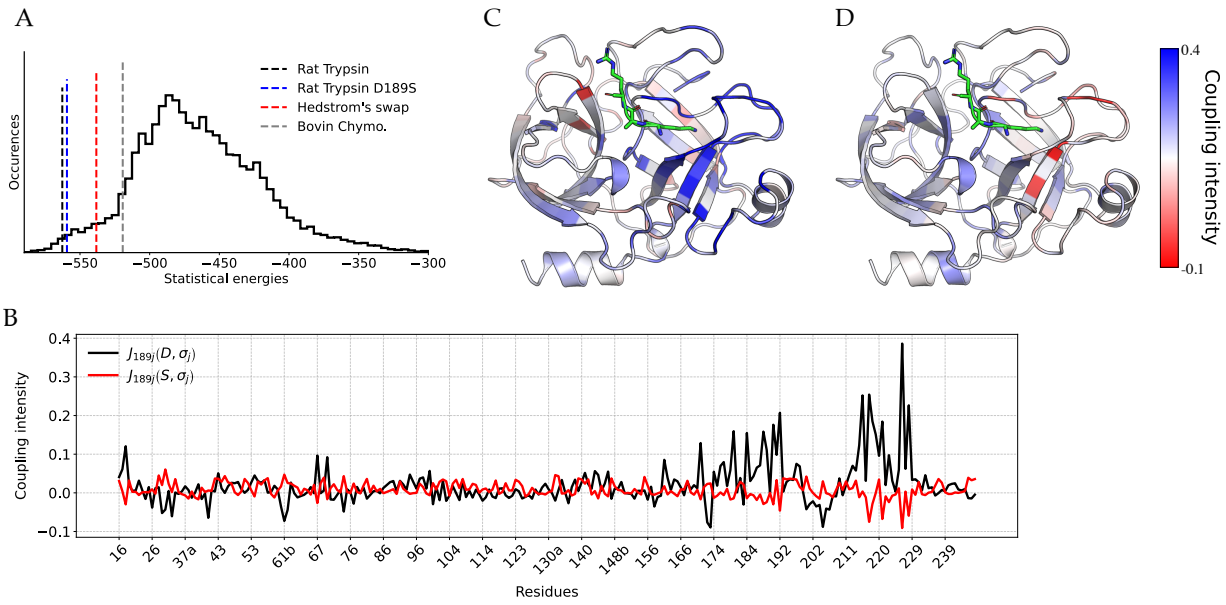

Figure 6.5: Statistical energy for a model inferred on the S1A family.
A. Distribution of statistical energies for natural sequences in the S1A alignment. Vertical dashed lines indicate the statistical energy of rat trypsin (black), its D189S mutant (blue), the Hedstrom swap (red), and bovine chymotrypsin (grey).
B. Coupling terms $J_{189j}(D,\sigma_j)$ (black) and $J_{189j}(S,\sigma_j)$ (red) across the sequence, showing how replacing D by S at position 189 affects coupling intensity with other sites.
C. Magnitude of the couplings $J_{189j}(D,\sigma_j)$ mapped onto the rat trypsin structure. Residues are colored from blue (strong positive coupling) to red (negative coupling). The green sticks highlight the substrate.
D. Same as in C but for the mutant $J_{189j}(S,\sigma_j)$, highlighting differences in coupling patterns due to the D189S mutation.

6.3 Statistical Energy

The DCA model is a Boltzmann distribution, with a statistical energy expressed as a sum of fields and couplings, each corresponding to statistical constraints:

$$E(\{\sigma_i\}_{i=1}^{L}) = -\sum_{i=1}^{L} h_i(\sigma_i) - \sum_{i<j} J_{ij}(\sigma_i,\sigma_j). \tag{6.1}$$

As illustrated in Figure 6.5A, the statistical energies of natural sequences are not all equal, reflecting differences in how well they conform to the statistics extracted from the MSA. However, considering the huge size of the sequence space, the chance of a random sequence falling within the same energy range as natural sequences is almost zero.

A negative coupling indicates that a combination of two amino acids at two specific sites is statistically unfavorable in the context of the studied family. However, this does not preclude the presence of this combination in sequences within the MSA.

On the other hand, it is absolutely possible to construct sequences with low energies yet lacking the activity of the natural enzymes. For instance, consider a wild-type sequence, such as the rat trypsin one. Introducing a mutation known to be deleterious in this context, like the D189S mutation [149], increases the sequence energy. The reason for this is that the mutation causes several residues that were previously positively coupled with residue 189 to become negatively coupled, as shown in Figure 6.5B. It is interesting to note that these residues are mostly located within the red sector, indicating the model's potential to

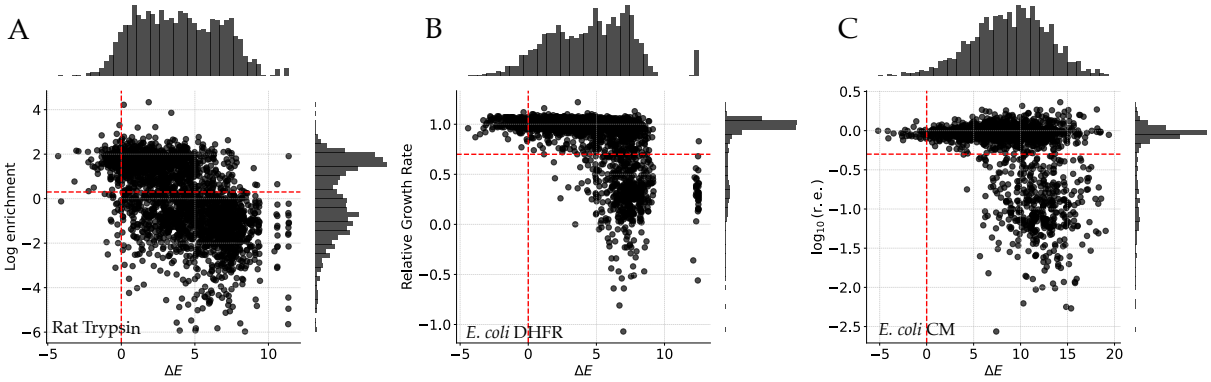

Figure 6.6: Comparison between statistical energy variations and DMS data.
Statistical energy changes (ΔE) are plotted against DMS measurements for single-point mutations in three proteins: rat trypsin [152] in **A**, DHFR [153] in **B**, and Chorismate Mutase [30] in **C**.
In all panels, a vertical red line separates mutations predicted to be beneficial ($\Delta E < 0$) from those predicted to be deleterious ($\Delta E > 0$). A horizontal red line indicates the boundary between enriched mutants (above) and non-enriched mutants (below).

identify interactions critical for the specificity of a serine protease (see Figure 6.5C & D). However, the D189S mutant cannot be distinguished from natural sequences based on this energy score. As an example, the Hedstrom swap and the bovine chymotrypsin sequences both exhibit higher energies.

In the end, the statistical energy score, while informative regarding the resemblance of a sequence to MSA-derived statistics, is the sum of numerous contributions[9] that may cancel each other out. As such, this score is not particularly reliable for guiding the prediction of compensatory mutations.

9: $L + \frac{L(L-1)}{2} = 33930$ contributions in this particular case.

6.4 Statistical energy variations

Statistical energy variations have been widely used in the literature to predict mutational effects, particularly for single-point mutations [75–77].

In the following, we define $\Delta E(\sigma_k : \mathrm{A} \rightarrow \mathrm{B})$ as the statistical energy variation associated with the mutation AkB in the sequence $\{\sigma_i\}_{i=1}^{L}$. This quantity depends on the local field at position k and the couplings between the mutated residue and the rest of the sequence:

$$\begin{aligned}\Delta E(\sigma_k : \mathrm{A} \rightarrow \mathrm{B}) &= E\left(\{\sigma_1, \ldots, \sigma_k = \mathrm{B}, \ldots, \sigma_L\}\right) - E\left(\{\sigma_1, \ldots, \sigma_k = \mathrm{A}, \ldots, \sigma_L\}\right) \\ &= -h_k(\mathrm{B}) - \sum_i J_{ki}(\mathrm{B}, \sigma_i) + h_k(\mathrm{A}) + \sum_i J_{ki}(\mathrm{A}, \sigma_i).\end{aligned} \tag{6.2}$$

A positive ΔE indicates that the mutation is predicted to be deleterious. However, since ΔE comprises multiple contributions, this prediction results from a combination of several terms.

Figure 6.6 compares ΔE values with experimental data from deep mutational scanning (DMS) studies. Across the three proteins, we find that very few mutations experimentally classified as deleterious are predicted to be beneficial by the model. Conversely, many mutations that are experimentally neutral are predicted as deleterious. This asymmetry can

be explained by revisiting the discussion in Chapter 3 of this thesis. For instance, we have shown that for the Chorismate Mutase family, over 68% of all possible amino acid combinations are absent from the dataset.

For a mutation AkB on the sequence σ, the couplings $J_{kj}(A, \sigma_j)$ may be negative, but there is at least one occurrence of each of these combinations in the data since the sequence σ itself is included. However, the same cannot be guaranteed for the couplings $J_{kj}(B, \sigma_j)$. It is highly likely that some of these combinations are missing from the data, causing the associated couplings to be strongly negative [10].

In the case of the rat trypsin DMS, it is important to note that selection is based on mutant activity against a trypsin substrate. Therefore, Figure 6.6A reflects the effect of mutations on trypsin-like activity, but says little about potential gains in chymotrypsin-like activity.

Let us now return to the example of rat trypsin, where we aim to identify compensatory mutations for D189S. Starting from the mutant rat trypsin D189S, we compute the statistical energy variation ΔE for all single-point mutations, select the most beneficial one (i.e., the mutation with the most negative ΔE, excluding position 189), and iterate this process on the newly obtained mutant.

Figure 6.7A highlights the residues involved in the first 20 beneficial mutations identified through this greedy procedure. These residues are broadly distributed across the protein and, notably, do not overlap with those in the red sector. Moreover, Figure 6.7B shows that all these mutations were already predicted as beneficial on the WT sequence; the D189S mutation has little to no impact on their ranking. Experimentally, these mutations also appear to have neutral effects on trypsin-like activity.

Since the only change between the WT and the D189S mutant is a single residue, the difference in ΔE is limited to terms involving interactions between position 189 and the mutated site which is insufficient to significantly alter the mutation ranking.

In conclusion, while statistical energy variation provides a useful first approximation of single-point mutational effect, it does not appear suited for identifying context-specific compensatory mutations. In the next section, we introduce another method designed to predict compensatory mutations with BM models.

10: It would be interesting to investigate how this asymmetry depends on the choice of the reference sequence used for predicting single-point mutation effects. One could explore this by removing sequences close to the reference from the training set.

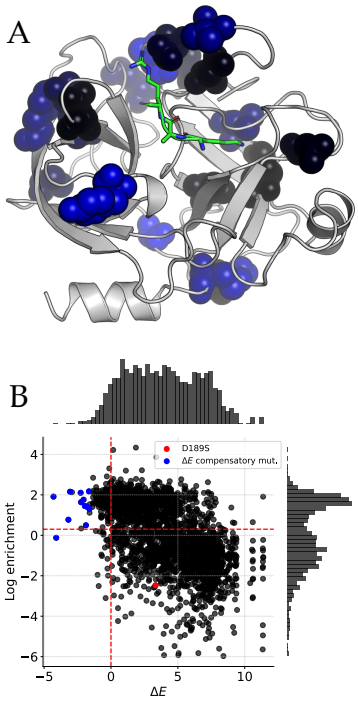

Figure 6.7: Compensatory mutations obtained via ΔE calculation.
A. Ribbon representation of rat trypsin (PDB: 3TGI) showing the first 20 most beneficial mutations, starting from the mutant rat trypsin D189S. The blue gradient indicates mutation order, from dark to light blue.
B. Same mutations plotted in the ΔE vs experimental enrichment plane, using DMS data from [152].

6.5 Predict compensatory mutations

As shown in the previous two sections, statistical energy and its variations, used as such, do not seem appropriate for predicting compensatory mutations. We therefore propose a method based on differences in statistical energy variations, that we will call COMBINE for COmpensatory Mutations via Boltzmann machine INference of Epistasis.

The general idea is to identify positive epistatic interactions, i.e. mutations that are predicted to be significantly more favorable, or less deleterious, in the context of a given mutant than in the wild-type background.

An illustration of the method is provided in Figure 6.8, using a minimal example for clarity. As a starting point, we consider a four-residue sequence denoted WT: GNDA. The aim is to identify compensatory mutations for the substitution σ_3 : D $\rightarrow$ S, also referred to as D3S (see Figure 6.8A).

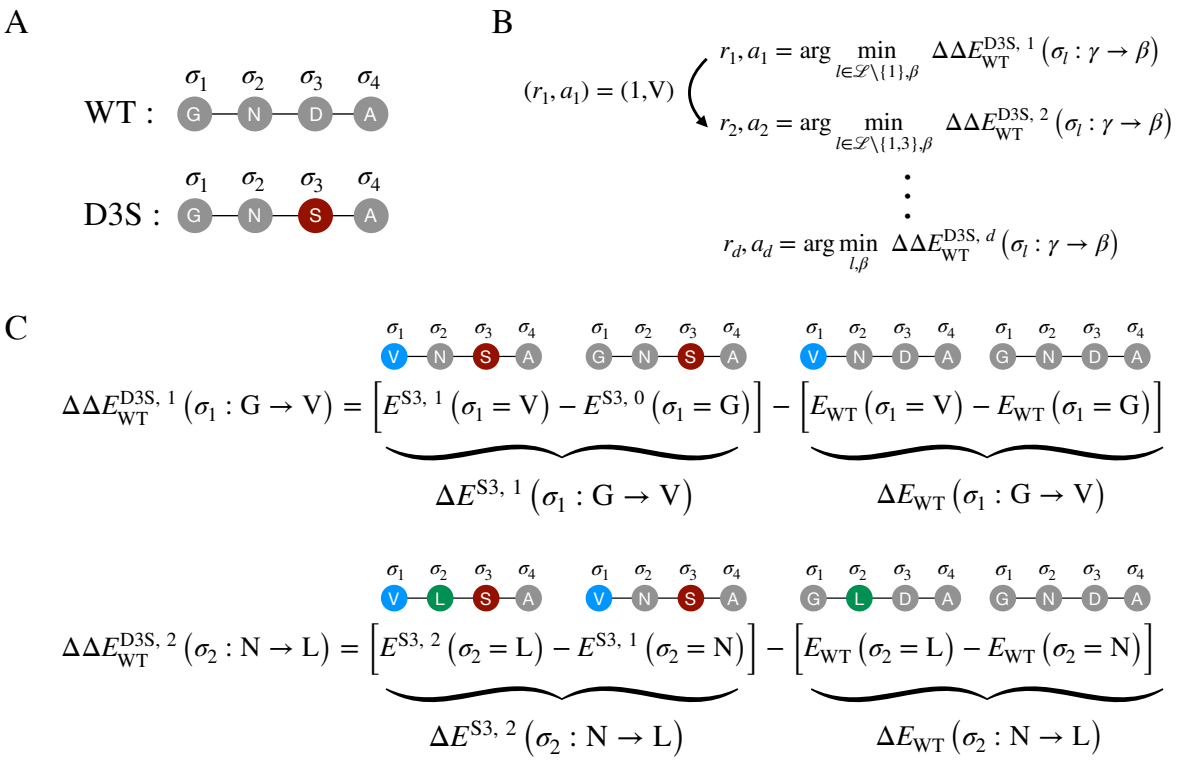

Figure 6.8: Example of a $\Delta\Delta E$ calculation
A. The starting wild-type sequence, WT, consists of four amino acids: GNDA. We want to find compensatory mutations for the D3S mutant: GNSA.
B. Successive compensatory mutations are selected by minimizing the quantity $\Delta\Delta E_{\mathrm{WT}}^{\mathrm{D3S},\,d}\left(\sigma_l : \gamma \to \beta\right)$ over all positions $l \in \mathscr{L}$ and all possible substitutions $\beta \in \mathscr{B}$. Here, the first compensatory mutation is assumed to be G1V.
C. The computation of $\Delta\Delta E_{\mathrm{WT}}^{\mathrm{D3S},\,1}\left(\sigma_1 : \mathrm{G} \to \mathrm{V}\right)$ is shown in the first row. It is written as the difference between two statistical energy variations: the effect of the mutation $\sigma_1 : \mathrm{G} \to \mathrm{V}$ in the S3 background (meaning on the mutant WT D3S), minus its effect in the WT background. Above each energy term, the corresponding sequence is shown, with residues that differ from WT highlighted in color.

We denote $\mathscr{L} := \{1, \ldots, L\}$ as the set of positions in the protein sequence, and $\mathscr{B} := \{-, \mathrm{A}, \ldots, \mathrm{Y}\}$ as the set of possible amino acids, including the gap character.

As shown in Figure 6.8B, our goal is to identify successive mutations that minimize the following quantity as we move away from the wild-type sequence:

$$\begin{aligned}\Delta\Delta E_{\mathrm{WT}}^{\mathrm{D3S},\,d}\left(\sigma_l : \gamma \to \beta\right) &= \Delta E^{\mathrm{S3},\,d}\left(\sigma_l : \gamma \to \beta\right) - \Delta E_{\mathrm{WT}}\left(\sigma_l : \gamma \to \beta\right) \\ &= \left[E^{\mathrm{S3},\,d}\left(\sigma_l = \beta\right) - E^{\mathrm{S3},\,d-1}\left(\sigma_l = \gamma\right)\right] \\ &\quad - \left[E_{\mathrm{WT}}\left(\sigma_l = \beta\right) - E_{\mathrm{WT}}\left(\sigma_l = \gamma\right)\right]. \qquad (6.3)\end{aligned}$$

Here, D3S is the initial mutation we aim to compensate, and WT is the reference wild-type sequence. The integer d denotes the number of mutations relative to WT, sequence positions are indexed by $l \in \mathscr{L} = \{1, 2, 3, 4\}$, and $\beta, \gamma \in \mathscr{B}$ represent amino acids.

The quantity $\Delta\Delta E_{\mathrm{WT_{ex}}}^{\mathrm{D3S},\,d}\left(\sigma_l : \gamma \to \beta\right)$ quantifies how much the mutation $\sigma_l : \gamma \to \beta$ is more beneficial—or less deleterious—when introduced on the D3S mutant with d compensatory mutations accumulated, compared to its effect in the wild-type background.

For $d = 1$, we compute $\Delta\Delta E_{\mathrm{WT}}^{\mathrm{D3S},\,1}\left(\sigma_l : \gamma \to \beta\right)$ for all positions $l \in \mathscr{L} \setminus \{3\}$ and all possible substitutions $\beta \in \mathscr{B}$.

An example of this computation for $l = 1$ and $\beta = \mathrm{V}$ is shown in Figure 6.8C. The first term, $\Delta E^{\mathrm{S3},\,1}(\sigma_1 : G \to V)$, measures the effect of mutating G to V in the background of the single mutant WT D3S. The second term, $\Delta E_{\mathrm{WT}}(\sigma_1 : G \to V)$, measures the effect of the same mutation in the wild-type sequence WT.

By minimizing $\Delta\Delta E_{\mathrm{WT}}^{\mathrm{D3S},\,1}$, we are thus searching for a mutation that is significantly more favorable—or less deleterious—in the context of the D3S mutant than in the wild type, indicating a positive epistatic interaction.

In terms of couplings, this quantity can be rewritten as:

$$\Delta\Delta E_{\mathrm{WT}}^{\mathrm{D3S},\,d}\left(\sigma_l : \gamma \to \beta\right) = -J_{3l}(\mathrm{S},\beta)+J_{3l}(\mathrm{S},\gamma)+J_{3l}(\mathrm{D},\beta)-J_{3l}(\mathrm{D},\gamma). \quad (6.4)$$

This expression shows that a highly negative $\Delta\Delta E_{\mathrm{WT}}^{\mathrm{D3S},\,1}$ typically corresponds to a residue $\sigma_l = \gamma$ that is negatively coupled to $\sigma_3 = \mathrm{S}$, and for which the substitution $\sigma_l : \gamma \to \beta$ introduces an amino acid β that is positively coupled to S.

Assuming the mutation minimizing $\Delta\Delta E_{\mathrm{WT}}^{\mathrm{D3S},\,d}\left(\sigma_l : \gamma \to \beta\right)$ is $\sigma_1 : \mathrm{G} \to \mathrm{V}$, we then search for a second compensatory mutation by minimizing $\Delta\Delta E_{\mathrm{WT}}^{\mathrm{D3S},\,2}\left(\sigma_l : \gamma \to \beta\right)$ across all positions $l \in \mathcal{L} \setminus \{1, 3\}$ and all substitutions $\beta \in \mathcal{B}$.

An example of this second-step calculation for $l = 2$ and $\beta = \mathrm{L}$ is also shown in Figure 6.8C. Here, we seek a mutation that is again more favorable—or less deleterious—in the background of the double mutant D3S G1V than in the wild-type sequence.

When written in terms of couplings, $\Delta\Delta E_{\mathrm{WT}}^{\mathrm{D3S},\,2}$ now involves not only interactions with site $l = 3$, but also with site $l = 1$, which was previously mutated to compensate for D3S.

By applying this procedure iteratively, we aim to identify successive mutations that compensate for the deleterious effects of those previously introduced.

6.5.1 First Application of COMBINE to rat trypsin D189S

We apply COMBINE to the rat trypsin sequence in the case where we aim to compensate for the mutation D189S. The quantity minimized iteratively is denoted as $\Delta\Delta E_{\mathrm{RatTryp}}^{\mathrm{D189S},\,d}$.

Mapping the first 20 compensatory mutations onto the rat trypsin structure reveals that all residues are located either directly on or near the Hedstrom loops (see Figure 6.9A). A more detailed analysis of this example is provided in Section 6.7, but before that, we provide a few clarifications on the methodology.

The method COMBINE identifies mutations that are predicted to exhibit positive epistatic interactions. However, these mutations may still be predicted as deleterious. In the case of rat trypsin D189S, most of the 20 compensatory mutations are in fact predicted to be deleterious in the context in which they are proposed (see quantity $\Delta E^{\mathrm{D189S},\,d}$ in Figure 6.9B). Here, we explain why we choose not to exclude such mutations.

In theory, if the inferred model could accurately predict mutational effects in any context, one could simply rely on the quantity ΔE, and the most

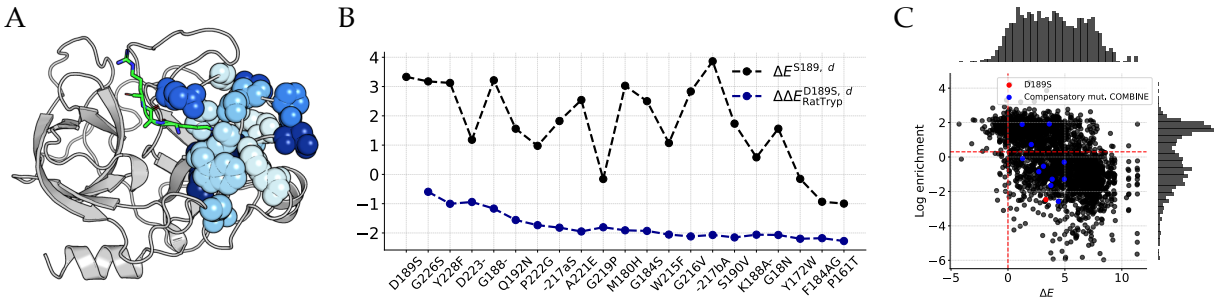

Figure 6.9: Compensating mutation D189S on rat trypsin.
A. Ribbon representation of the rat trypsin structure (PDB: 3TGI), showing the first 20 compensatory mutations obtained with COMBINE starting from the D189S mutant. The blue gradient indicates the mutation order from dark to light.
B. Minimum value of $\Delta\Delta E^{\text{D189S, } d}_{\text{RatTryp}}$ (blue) as a function of distance from the WT. The corresponding mutations are shown on the x-axis, and their statistical energy variation $\Delta E^{\text{S189, } d}$ is plotted in black.
C. Mutations of the panel A shown in the 2D space defined by model-predicted ΔE and experimental log enrichment from [152].

beneficial mutation in the rat trypsin D189S background would naturally be one that compensates D189S.

However, as discussed in Section 6.4, this does not seem to hold true. Even in the WT background, the model often predicts many neutral mutations as deleterious, largely because unobserved combinations in the data are penalized. Although mutational scan data for the D189S mutant are not available, it is likely that a similar pattern would be observed in this background, as well as in multiple mutants further away from the WT.

Figure 6.9C shows the effect of each of the 20 successive compensatory mutations on trypsin activity, measured in the rat trypsin background (for which mutational scan data are available [152, 154]). Many of these mutations are predicted as deleterious by the model ($\Delta E < 0$), and some are also experimentally confirmed to reduce trypsin activity.

This does not tell us how these mutations affect chymotrypsin activity in the successive mutants as we move away from the WT sequence. However, it does illustrate a key difference from the ΔE-based analysis in Section 6.4, which only selected mutations predicted to have neutral or beneficial effects on trypsin activity in the WT context.

Take, for instance, the mutation G226S which is the first proposed compensatory mutation for D189S. It is predicted as deleterious both by the model and by experimental mutational scan for trypsin activity in rat trypsin. According to Figure 6.9B, G226S is also predicted to be deleterious in the background of the single mutant rat trypsin D189S.

When computing $\Delta\Delta E^{\text{D189S, } 1}_{\text{RatTryp}}$, however, we choose to ignore this and do not consider any context beyond position 189. While this may seem drastic, it is supported by studies suggesting that although epistasis is pervasive [125, 133], the network of epistatic interactions is also remarkably sparse [132].

We therefore restrict ourselves to identifying residues that show statistically unfavorable interactions with previously mutated sites and for which a substitution would be statistically favorable, without considering the broader impact on the rest of the sequence.

In the following sections, we will apply COMBINE in two different scenarios:

- Trypsin-to-chymotrypsin conversion initiated by a mutation at position 189

- Chymotrypsin-to-trypsin conversion initiated by a mutation at position 189

In each case, we compare our results to the literature and present preliminary experimental data. In the following section, we provide additional context to help interpret the experimental results.

6.6 Experimental testing of protease activity

As discussed in Section 5.1.3, it is a common practice to measure the ratio $\frac{k_{cat}}{K_m}$ to compare the catalytic activity of a single enzyme across different substrates (i.e., specificity), or to compare different enzymes on their respective substrates.

For the S1A family, the catalytic activity corresponds to the enzyme's ability to cleave a peptide bond within a given substrate. To assess this, a substrate can be engineered with a fluorophore that becomes fluorescent upon cleavage [155].

By mixing a solution containing the enzymes with one containing the substrates and monitoring fluorescence over time, the enzymatic activity can be estimated (see Figure 6.10 for a schematic illustration of the experimental principle). Fluorescence is typically reported in Relative Fluorescence Units (RFU), an arbitrary unit proportional to the intensity of the emitted signal.

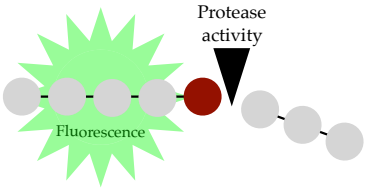

Figure 6.10: Schematic representation of S1A sequence testing.
The substrate, depicted in grey (each amino acid represented by a grey circle), is engineered to become fluorescent upon cleavage by the enzyme, shown as a black triangle. The amino acid immediately upstream of the cleavage site, referred to as P1, is highlighted in red.

The fluorescence curves shown below illustrate these time-course measurements. These preliminary results stem from experiments carried out by Amaury Paveyranne, Timothé Lucas, and Shoichi Yip at the Laboratoire Jean Perrin.[11].

11: For technical details on the experimental assays, the interested reader is referred to Amaury Paveyranne's thesis [152]. Additional information is also provided in Appendix D.

Before proceeding, it is worth highlighting that:

- According to the Michaelis–Menten model, the initial slope of the fluorescence curve, which corresponds to the initial rate of product formation, is given by: $k_{\text{cat}} \frac{[\text{E}]_{tot}[\text{S}]_0}{K_m + [\text{S}]_0}$. Here, $[\text{E}]_{\text{tot}}$ denotes the total enzyme concentration (including enzyme–substrate complexes), and $[\text{S}]_0$ is the initial substrate concentration. While this value depends on both the turnover number $k$cat and the Michaelis constant K_m, it does not, on its own, allow for the determination of the catalytic efficiency ratio $\frac{k_{\text{cat}}}{K_m}$ across the tested mutants. Indeed, this would require measurements at varying initial substrate concentrations with a controlled enzyme concentration (achievable through purification).
- The fluorescence curve obtained for a given sequence depends on both the enzyme and substrate concentrations in the solution. While substrate concentrations are assumed to be roughly the same across experiments, this is not the case for enzyme concentrations. As a result, fluorescence curves can only be reliably compared across variants if their expression levels are similar. In what follows, we report expression levels for each tested sequence in Relative Fluorescence Units (RFU).
- Replicates are available for some measurements, and confidence intervals have been added to certain curves.

6.7 Trypsin to chymotrypsin conversion

As previously explained, the first successful conversion was achieved by Hedstrom *et al.* in the early 1990s. Starting from rat trypsin and mutating the 16 residues highlighted in Figure 6.2, they obtained an enzyme displaying between 2% and 15% of bovine chymotrypsin's $\frac{k_{cat}}{K_m}$ on chymotrypsin substrates.

In this work, we chose to use the same rat trypsin sequence as a starting point, introduced the D189S or D189G mutation (as informed by the analyses in Section 6.2) and applied the COMBINE methodology described in Section 6.5.

Figure 6.11A compares the positions of the top 20 compensatory mutations starting from D189S with the residues involved in Hedstrom's swap. While not all proposed mutations are part of Hedstrom's swap, nearly all are located near the corresponding loops. Unsurprisingly, their positions on the structure illustrated in Figure 6.11B are consistent with those in Figure 6.2.

Starting from D189S, the mutants proposed by the COMBINE method are labeled as rat trypsin D189S X, where X indicates the number of mutations relative to the rat trypsin wild type. We tested: rat trypsin D189S 3 (also noted as "D189S G226S Y228F"), rat trypsin D189S 5, rat trypsin D189S 7, rat trypsin D189S 9, and rat trypsin D189S 14.

From D189G, we tested rat trypsin D189G 3 (also noted as "D189G G226D Y228F") and rat trypsin D189G 8.

Additionally, we tested a triple mutant starting from the first compensatory mutation proposed for rat trypsin D189S, which results in the mutant G226S D189G Y228F. The complete list of all tested mutants and with the details of the mutations is provided in Appendix D.

Fluorescence curves measured on a "chymotrypsin" substrate are shown in Figure 6.11D & E. Besides the mutants, we include bovine chymotrypsin as a positive control, rat trypsin as a negative control, and Hedstrom's swap (these three curves appear on both figures).

Fluorescence for bovine chymotrypsin rapidly reaches about 10^3 RFU, whereas rat trypsin barely exceeds 10 RFU, consistent with the known specificities of these natural sequences.

The result for Hedstrom's swap (with three replicates) may seem surprising, as its activity appears to surpass bovine chymotrypsin. However, as already mentionned these curves are not directly comparable since bovine chymotrypsin expression levels are on average four to five times lower than those of Hedstrom's swap in these experiments.

Among the mutants proposed by COMBINE starting from rat trypsin D189S, only the triple mutant D189S G226S Y228F displays a detectable, albeit modest, activity on the chymotrypsin substrate. The other variants show no significant activity compared to the wild-type rat trypsin, but their low expression levels may limit interpretation. While it is clear that the Hedstrom swap exhibits chymotrypsin-like activity, further experiments are required to draw definitive conclusions about the other mutants.

As shown in Figure 6.11E, starting from the D189G background, the fluorescence curve for the triple mutant D189G G226D Y228F exhibits a marked increase, exceeding 10^2 RFU, with expression levels about twofold lower than those of the Hedstrom swap. The triple mutant G226S

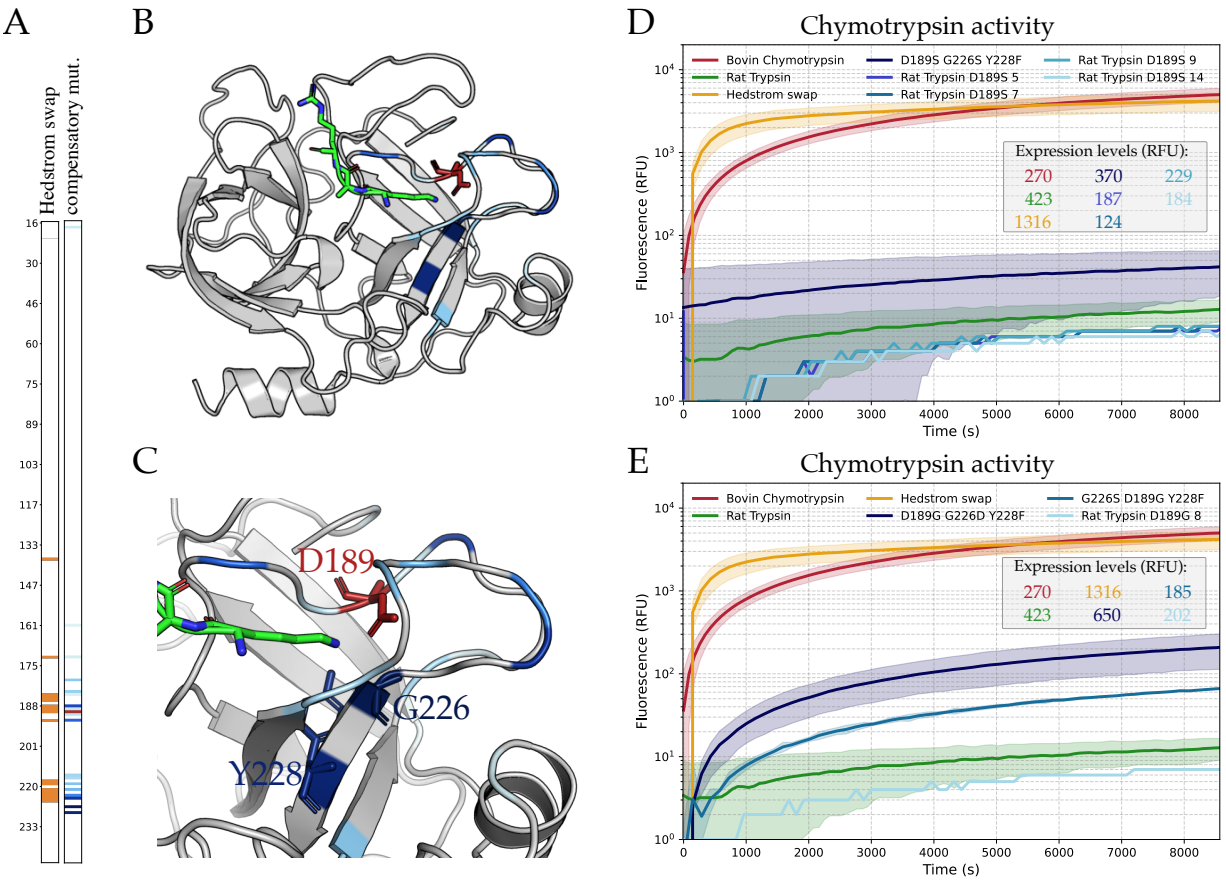

Figure 6.11: Rat trypsin to chymotrypsin conversion. (Panels D & E use data provided by Amaury P.)
A. The two vertical bars represent an S1A sequence, with residues mutated in Hedstrom's swap highlighted in orange on the left, and compensatory mutations proposed by the COMBINE method (starting from rat trypsin D189S) on the right. The gradient indicates the order in which mutations are proposed, from dark to light blue. Residue numbering follows the PDB 3TGI sequence.
B. Ribbon representation of rat trypsin (PDB: 3TGI), highlighting the first 20 compensatory mutations from the D189S mutant, using the same color gradient as in panel A. Residue D189 is shown in red.
C. Zoomed-in view of the binding pocket. Residue D189 is highlighted in red, with sticks representing the first two residues proposed for compensatory mutations: G226 and Y228.
D. Fluorescence curves over time for natural sequences and mutants tested on a chymotrypsin substrate. Bovine chymotrypsin (red) serves as positive control, while rat trypsin (green) is the negative control. The orange curve corresponds to Hedstrom's swap, and the blue curves represent successive mutants proposed by COMBINE. The color gradient reflects proximity to the rat trypsin WT sequence: from dark blue for the triple mutant rat trypsin D189S G226S Y228F, to light blue for rat trypsin D189S 14, which carries 14 mutations relative to WT. Average expression levels (measure of fluorescence (RFU)) of the tested sequences are indicated on the graph in the same order and with the same colors as in the legend.
E. Same as panel D, but the blue curves correspond to two triple mutants starting from rat trypsin D189G and rat trypsin G226S, as well as rat trypsin D189G 8, which accumulates eight mutations relative to WT to compensate for D189G.

D189G Y228F, despite being weakly expressed (185 RFU), also appears to display significant activity on the chymotrypsin substrate. In contrast, rat trypsin D189G 8 shows no measurable activity under these conditions.

All these mutants were also tested on a "trypsin" substrate, and fluorescence data indicate a loss of trypsin activity for all. Interested readers can find details in Appendix D.

While these experimental results are preliminary and require further investigation—especially to address disparities in expression levels—they nonetheless suggest that the proposed triple mutants exhibit chymotrypsin-like activity. To our knowledge, this is the first evidence that a chymotrypsin activity can be achieved with so few mutations to rat trypsin. The following section presents a more detailed analysis of these mutants.

6.7.1 Focus on the triple mutants

The same three residues—189, 226, and 228—are involved in all three triple mutants that exhibit measurable activity in fluorescence-based assays. These residues are part of the binding pocket and, as shown in Figure 6.11C, are therefore located in close proximity to the substrate.

The fact that just three mutations within the binding pocket are sufficient to confer chymotrypsin-like activity is a noteworthy result. Indeed, apart from the Hedstrom swap, which involves 16 mutations, no mutant closer to rat trypsin has been reported to display chymotrypsin activity.

Additionally, a Statistical Coupling Analysis (SCA) performed on the MSA used in this study also identifies these three positions as the top contributors to the component associated with the specificity sector (see Figure 6.3). This suggests that the strongest couplings involving residue 189 capture similar information to that revealed by the sector analysis.

The D189S mutation is part of the Hedstrom swap, whereas positions 226 and 228 are not. Indeed, Hedstrom *et al.* performed their swap between rat trypsin and bovine chymotrypsin, which share the same amino acids at positions 226 and 228.

In contrast, the COMBINE method is not restricted to substitutions observed between rat trypsin and bovine chymotrypsin. The only constraints are the initial sequence and the first mutation to be compensated.

As a result, there is no guarantee that compensating the D189S mutation will necessarily shift specificity toward chymotrypsin-like activity. For example, the alignment contains fewer than 40 sequences annotated as elastases, yet 60% of them carry a serine at position 189. By imposing the D189S mutation as a starting point, there is a possibility of shifting toward elastase-like specificity. We therefore tested the mutants on an elastase substrate, but fluorescence measurements showed no significant activity.

In Figure 6.12, we examine the sequence identity between rat trypsin and sequences from the alignment that carry combinations suggested by COMBINE. The vast majority of these sequences share less than 40% identity with rat trypsin. The wild type background in which we introduced these mutations is thus quite different from the natural context in which these combinations are typically found[12].

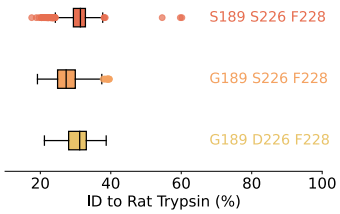

Figure 6.12: Sequence identity to rat trypsin for sequences carrying combinations proposed with COMBINE. The multiple sequence alignment (MSA) used in this study was filtered to retain only sequences carrying the amino acid substitutions proposed by the COMBINE method. The box plots show the percentage identity of these sequences to rat trypsin, for each of the three combinations: S189 S226 F228 (top), G189 S226 F228 (middle), and G189 D226 F228 (bottom).

12: Unfortunately, most of these sequences lack specificity annotations. For instance, the alignment contains 1287 sequences with the combination S189 S226 F228, but only 18 of them are annotated—11 of which are labeled as chymotrypsins.

6.8 Chymotrypsin to trypsin conversion

To our knowledge, no successful conversion of a chymotrypsin into a trypsin has been reported to date. However, several attempts have been made. In 1996, Venekei *et al.* tested the reverse of Hedstrom's swap using bovine chymotrypsin, but without success [145] (see Table 6.1). Later, in 2004, Jelinek *et al.* showed that the double mutant rat chymotrypsin S189D A226G exhibited a weak trypsin-like activity, estimated at just 0.01% of that of wild-type trypsin.

We applied the COMBINE method to both bovine and rat chymotrypsin, using S189D as the imposed mutation. Here, we present the results obtained for the rat chymotrypsin background.

Figure 6.13 summarizes the outcomes. Panel A shows a comparison between the mutations proposed by COMBINE and those introduced

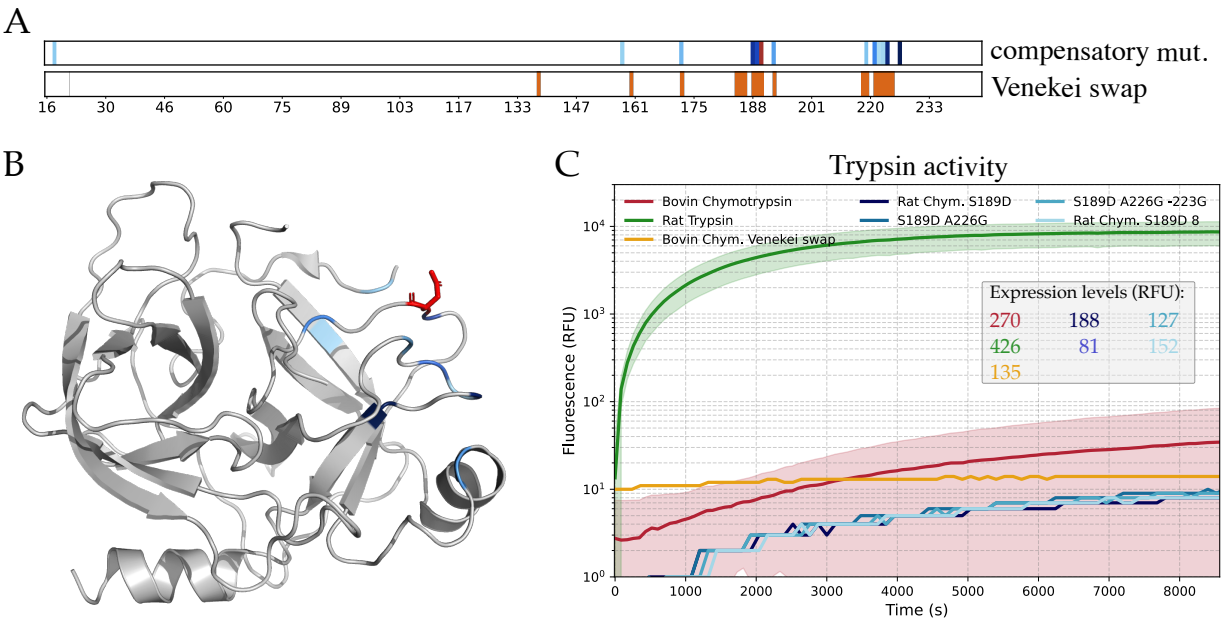

Figure 6.13: Rat chymotrypsin to Trypsin Conversion (Panel C uses data provided by Amaury P.)
A. The two horizontal bars represent an S1A sequence. Compensatory mutations proposed by the method (starting from rat chymotrypsin S189D) are shown above, while the Venekei swap mutations are highlighted in orange below. A blue color gradient indicates the order in which the compensatory mutations were proposed, from dark to light blue. Residue numbering is based on the PDB 3TGI sequence.
B. Ribbon representation of rat chymotrypsin (PDB: 1KDQ), highlighting the first 20 compensatory mutations proposed from the S189D mutant. The same color gradient as in panel A is used, and residue 189 is shown in red.
C. Fluorescence time-course curves for natural and mutant sequences tested on a trypsin substrate. Rat trypsin (green) serves as a positive control, while bovine chymotrypsin (red) serves as a negative control. The orange curve corresponds to the Venekei swap applied to bovine chymotrypsin. The dark blue curve shows the single mutant rat chymotrypsin S189D, and lighter blue curves represent mutants proposed by the COMBINE method. A color gradient reflects their sequence similarity to wild-type rat chymotrypsin. Average expression levels are indicated next to each curve.

in the Venekei swap. As seen in the reverse conversion (trypsin to chymotrypsin), the proposed mutations largely overlap with the Hedstrom loops. Panel B shows their spatial distribution in the structure of rat chymotrypsin.

Strikingly, the first mutation proposed to compensate for S189D is A226G, which is the exact substitution introduced by Jelinek *et al.* We therefore experimentally tested this double mutant, as well as a triple mutant S189D A226G G223G, and rat chymotrypsin S189D 8 (which accumulates seven compensatory mutations on top of S189D).

Panel C of Figure 6.13 shows fluorescence measurements on a trypsin-specific substrate. Rat trypsin rapidly reaches fluorescence levels of approximately 10^3 RFU, while bovine chymotrypsin remains below 10 RFU—consistent with the expected specificities of these enzymes.

All other tested mutants—whether reported in the literature or proposed via COMBINE—exhibit fluorescence signals lower than that of bovine chymotrypsin on the trypsin substrate[13]. Nonetheless, as in previous experiments, expression levels for these constructs are relatively low.

Taken together with previous studies, these findings suggest that acquiring trypsin-like activity through a small number of mutations on a chymotrypsin sequence may be more challenging than the reverse conversion. However, the underlying reasons for this asymmetry remain unclear.

13: For the double mutant reported by Jelinek *et al.*, it is possible that the activity—measured at only 0.01% of that of wild-type trypsin—is too low to be detected with our experimental setup.

6.9 Back to Regularization

In Chapter 3 and Chapter 4, we extensively discussed the importance of regularization in BM models.

We particularly emphasized that regularization should be adapted to the task at hand. For instance, in the context of contact prediction from couplings, a regularization scheme that introduces a bias in favor of pairwise interactions is beneficial. However, such regularization may not be suitable for generative modeling.

In the present section, we thus discuss the impact of regularization on the prediction of compensatory mutations using the COMBINE method.

In the scenario studied here—where the mutation to be compensated lies within a sector that is robustly identified in the literature [19, 29, 36]—it seems reasonable to expect that the model should capture the coevolutionary signal associated with this sector.

This expectation is indeed met in the results presented in Section 6.7 and Section 6.8, where the majority of proposed compensatory mutations fall within the red sector.

But how does the applied regularization affect these results?

Interestingly, we observe that the outcomes are quite robust regardless of the model used (BM or SBM) and across a range of regularization strengths[14] (see Figure D.4 for the detailed results). This may seem surprising in light of the arguments made throughout the second part of this thesis.

14: Except in cases of excessively strong or weak regularization, which inevitably deteriorate the predictions.

However, this robustness can be understood as follows: the COMBINE method relies on a computation that involves only a small subset of coupling terms in the prediction.

For example, consider the first compensatory mutation in the rat trypsin D189S mutant. The quantity we aim to minimize is:

$$\Delta\Delta E^{\mathrm{D189S},\,1}_{\mathrm{RatTryp}}\left(\sigma_l : \gamma \to \beta\right) = -J_{189l}(\mathrm{S},\beta)+J_{189l}(\mathrm{S},\gamma)+J_{189l}(\mathrm{D},\beta)-J_{189l}(\mathrm{D},\gamma). \tag{6.5}$$

This expression involves only couplings related to residue 189. Thus, minimizing this quantity effectively amounts to comparing the couplings between residue 189 and other residues in the sequence.

In this context, what matters is the relative ranking of the couplings involving the residue 189. This stands in contrast to contact prediction, where all couplings are ranked globally, and the hope is that the strongest ones correspond to residue pairs in physical contact.

6.10 Summary and conclusions

Proteins in the S1A family are enzymes that catalyze the hydrolysis of peptide bonds with diverse specificities. Many studies have explored the mechanisms underlying this functional diversity, often aiming to convert the specificity of these enzymes using a limited number of mutations.

We began by discussing the challenges of this task through examples from the literature, focusing in particular on the work of Hedstrom *et*

al., who in 1994 demonstrated that a chymotrypsin-like activity could be achieved by introducing 16 mutations into the rat trypsin sequence. Their findings highlighted that, although residue 189 plays a critical role in specificity, it is not sufficient on its own to induce a switch.

Building on this insight, we proposed to take a mutation at this key residue as a starting point and to use a BM model trained on the S1A family to predict compensatory mutations that could shift the enzyme's specificity.

After showing that ΔE is not a suitable metric for this type of prediction, we introduced the COMBINE method, which identifies mutations predicted to have a positive epistatic coupling with the mutation to be compensated.

We applied this approach in two different scenarios and provided preliminary experimental results.

In the case of the trypsin-to-chymotrypsin conversion, we identified three triple mutants that exhibit detectable chymotrypsin-like activity. Although their activity remains lower than that measured for the 16-residue Hedstrom swap in fluorescence-based assays, differences in expression levels make direct comparisons difficult.

The key result of this study is that only three mutations, located within the binding pocket, are sufficient to obtain a significant chymotrypsin activity starting from rat trypsin. Interestingly, two of the three mutated positions are absent from the Hedstrom swap but belong to the specificity sector identified through SCA. These findings may encourage the application of directed evolution[15] methods to investigate potential conversion pathways starting, for example, from rat trypsin D189S.

15: Directed evolution is a method that mimics natural selection in the laboratory to evolve proteins with desired properties through iterative rounds of mutation and selection.

In the reverse direction—from chymotrypsin to trypsin—none of the COMBINE-predicted mutants displayed significant trypsin activity. This mirrors previous findings in the literature and reinforces the observation that acquiring trypsin-like activity through a few mutations on a chymotrypsin scaffold is more challenging.

6.10.1 Future Directions

First, the results on the conversion of rat trypsin to chymotrypsin are promising and suggest several directions for further investigation. A natural next step would be to take a closer look at the triple mutants that showed detectable chymotrypsin-like activity. In particular, it would be useful to test all single mutants and all pairwise combinations of the mutations found in the triple mutants, to see whether some of these variants already show chymotrypsin-like activity. Beyond the kinetic curves presented in this chapter, measuring the $k_{\mathrm{cat}}/K_{\mathrm{m}}$ ratio would be valuable for comparing our results with those reported in the literature.

It would also be interesting to explore conversions starting from other wild-type sequences—whether trypsin from different organisms or proteases with other specificities.

For example, we examined the trypsin-to-thrombin conversion, though the results are not included in this manuscript. In this case, our analysis was limited to a comparison with previously published results. Nonetheless, we observed that the mutations proposed were consistent with those introduced by Page *et al.* in 2006 [150].

As for the COMBINE method itself, it would be useful to study in more detail the conditions under which it can make reliable predictions.

First, the model clearly requires that the mutation to be compensated is represented in the sequence data. For example, it would be unrealistic to predict a compensation for the D189S mutation if no sequence in the MSA naturally carries a serine at position 189. More generally, predictions are more likely to be meaningful when the sequence being studied is well represented in the alignment—in other words, when it's not too far from the typical background of the MSA.

In the context of specificity conversion, we also need at least one initial mutation[16] to define a starting point for the trajectory. If reliable specificity annotations were available—which is not yet the case for the S1A family—we could imagine extending the input MSA with a pseudo-residue encoding specificity and training the model accordingly. The first mutation to be compensated would then correspond to a change in this label residue.

16: Though one could consider starting from multiple mutations instead.

Currently, the method selects mutations with the strongest predicted epistatic interactions. But other strategies could be explored. For example, we considered using Monte Carlo sampling to propose mutational trajectories starting from a variant like rat trypsin D189S. By adding a penalty term in the Hamiltonian for distance to the starting sequence, one could explore different candidate conversion paths within a defined mutational radius.

Lastly, it would be interesting to apply COMBINE to other protein systems. This is precisely what we will do in the next chapter, focusing on the dihydrofolate reductase family.

Compensatory mutation within allosteric network of *E. coli* DHFR

7

The present chapter focuses on a study that also involve Paul Guenon, Damien Laage and Guillaume Stirnemann at ENS, Karolina Filipowska and Kim Reynolds from the University of Texas, and Clément Nizak from the Jean Perrin laboratory.

Paul Guenon, a PhD student co-supervised by Damien Laage and Olivier Rivoire, focuses his research on allosteric communication in proteins, using molecular dynamics simulations as a primary tool. As part of this project, he studied dihydrofolate reductase (DHFR), an enzyme in which a mutation distant from the active site is known to severely reduce catalytic activity.

My contribution to this project was to treat this long-range mutation as one requiring compensation and to predict additional mutations that could restore the enzyme activity.

Meanwhile, Karolina Filipowska and Kim Reynolds carried out experimental assays to measure the catalytic activity of both the wild-type DHFR and the mutant, providing data to validate our predictions.

7.1 Allostery

Allostery is an essential property of proteins, characterized by the thermodynamic coupling of distant sites without direct physical interaction. The spatial separation between these sites implies the existence of a communication mechanism that transmits information across the protein structure [39, 156].

As illustrated in Figure 7.1, the binding of a ligand at one site can for example induce conformational changes that modulate the activity of a distant site, a principle known as allosteric regulation. This mechanism is not limited to enzymes [157] but extends to a wide range of proteins, including for example structural proteins [158], and molecular motors [159].

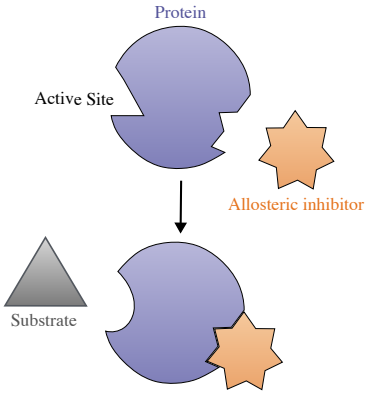

Figure 7.1: Allosteric regulation. Schematic example of an allosteric regulation where the binding of an inhibitor modifies the active site of the protein so that subtrate binding is prevented.

Several theoretical models have been developed to describe allostery, including the classic Monod-Wyman-Changeux (MWC) model [160], in which ligand binding induces a shift in the equilibrium between two preexisting conformational states with distinct activities. In contrast, the Koshland-Nemethy-Filmer model [161] proposes that ligand binding triggers a structural change toward a new conformation not previously populated.

Evidence suggest that allostery is a pervasive property of proteins, meaning that nearly all proteins possess inherent allosteric features latently encoded within the sequence [162]. In what follows, we will focus on dihydrofolate reductase (DHFR), an enzyme that has been extensively studied for its allosteric properties. For instance, SCA combined to experimental measurements revealed the presence of a sector connecting the active site to specific surface residues acting as hot spots for the emergence of allosteric control [36].

7.2 Dihydrofolate reductase

1: This molecule belongs to the folate family, whose members are derived from vitamin B9.

Dihydrofolate reductase (DHFR) is an enzyme that, as its name suggests, catalyzes a reduction reaction. Specifically, it transfers hydrogen atoms and electrons to a molecule known as dihydrofolate (H_2F), thereby converting it into tetrahydrofolate (H_4F)[1].

The structure of *E. coli* DHFR, illustrated in Figure 7.2, includes the subtrate H_2F, as well as NADPH, which acts as a cofactor participating in the chemical reaction. In the following we will also mention another molecule, H_3F, which is a protonated intermediate of the substrate. Indeed, the second step of the reaction involves the transfer of a hydride ion (H^-) between H_3F and NADPH. The rate constant of this hydride transfer, k_{hyd}, determines the efficiency of this reaction step.

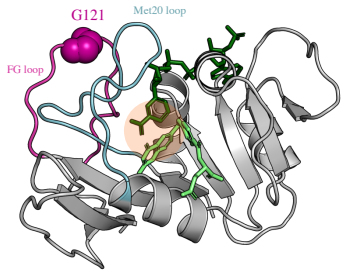

Figure 7.2: Ribbon representation of the *E. coli* DHFR structure (PDB: 1RX2). The enzyme is shown in grey cartoon representation. Two bound molecules are depicted: the substrate dihydrofolate (H_2F) in light green, and the cofactor NADPH in dark green. The FG loop is colored magenta, with residue G121 shown as spheres, while the Met20 loop is highlighted in light blue. The location of the active site is indicated by an orange circle.

Tetrahydrofolate, the reaction product, is essential for nucleic acid synthesis and, by extension, for cell growth. Given its key role in cellular metabolism, DHFR has emerged as a major target for drugs designed to prevent cell proliferation. For instance, DHFR inhibitors are widely used in the treatment of infections and cancer [163].

Because of its metabolic role and therapeutic relevance, DHFR has been the focus of extensive research. Among the studies conducted, attention has been paid to the enzyme's allosteric properties, offering deeper insights into its regulation and function.

7.2.1 Allostery in DHFR

In the late 1990s, some studies began to focus on a particular residue located over 13 Å from the active site[2] of *E. coli* DHFR. As illustrated in Figure 7.2, this glycine residue—designated as G121—is positioned at site 121 on the enzyme's FG loop. Experimental work showed that substituting this glycine (G) with a valine (V) resulted in a dramatic ~99.4% decrease in the hydride transfer rate between H3F and the cofactor NADPH [164][3].

2: For comparison, allosteric effects are typically considered beyond a threshold of ~ 10 Å , which is about twice the typical distance between two residues in direct contact.

3: Other substitutions, including G121A, G121S, G121L, and G121P, were also tested, and the effect on k_{hyd} correlated with the size of the amino acid side chain replacing glycine.

These findings highlighted the crucial role of residue 121 in the catalytic process. In particular, they suggested that G121, despite its distance from the active site, is functionally coupled to it, thus revealing an underlying allosteric mechanism within DHFR. Following this discovery, further investigations confirmed the allosteric importance of G121 [165, 166], although the precise molecular pathways mediating this long-range communication remained unknown.

To better understand the structural basis of properties like allostery, methods analyzing amino acid coevolution—such as SCA—can be employed. As discussed in Chapter 5, functionally relevant couplings between residues, regardless of the specific physical mechanism, are indeed expected to drive coevolution. Applying SCA to DHFR, Reynolds *et al.* [36] identified a sector forming a physically contiguous network that connects the active site to several distant surface regions. Notably, residue 121—our focus here—was predicted to be part of this sector.

Given that Molecular Dynamics (MD) simulations provide a powerful framework for probing the dynamic behavior of molecular systems, they have been extensively used to study motions linked to allosteric transitions across various proteins [167, 168]. Building on this approach, Paul Guenon thus conducted MD simulations to explore the effects of the G121V mutation and to propose a molecular mechanism for allosteric communication in DHFR.

7.3 Molecular dynamics simulations

In this section, we briefly summarize some of the key findings of the study conducted by Paul Guenon.

Molecular dynamics (MD) simulations provide a framework for modeling the motions of biological molecules over time, offering insights into how proteins explore their conformational space.

However, this task is far from straightforward, as the timescales associated with significant conformational changes can be long, and systems can become trapped in metastable states. To facilitate the exploration of the conformational space, Paul G. and his supervisors thus employed methods such as Replica Exchange Molecular Dynamics (REMD) [169]. In this technique, multiple simulations (replicas) are run in parallel at different temperatures, allowing for periodic exchanges between them with a Metropolis criterion to overcome energy barriers.

7.3.1 Conformational space of *E. coli* DHFR and mutants

Starting from the structure shown in Figure 7.2, REMD simulations were performed to explore the conformational landscape of the *E. coli* DHFR, as well as several of its mutants.

In the case of DHFR, these simulations revealed two distinct conformations that were already present for the wild-type sequence. A more detailed analysis further showed that the relative orientation between the cofactor and the substrate significantly differs between these two conformations[4]. One conformation features a parallel alignment, while the other exhibits a perpendicular arrangement between the cofactor and the substrate.

The distribution of conformations based on this orientation criterion is shown in Figure 7.3A for three different sequences: the wild-type (WT), and two mutants, G121V and G121A, that have already been studied in the literature [164]. In particular, G121A is known to exhibit an enzymatic activity intermediate between that of the WT and G121V mutant.

Although the conformational landscapes differ significantly across these three sequences, in each case the space remains clearly partitioned into two dominant modes. For the WT, the parallel conformation is far more populated than the perpendicular one, while the G121V mutant shows an almost complete shift towards the perpendicular state. As for G121A, the two conformations are populated at roughly equal levels.

An example of a parallel and perpendicular conformation is respectively shown in Figure 7.7A & B. Beyond the distinct orientations of the cofactor and substrate, we observe that the 2 loops also adopt different conformations. Notably, the increased separation between the loops in the perpendicular conformation arises from a steric hindrance, which becomes more even pronounced when the residue at site 121 has a longer side chain. The connection between structural perturbations in the loops and changes in cofactor–substrate orientation is elucidated by the allosteric mechanism proposed by Paul Guenon, which will not be detailed here.

Importantly, the functional significance of these conformations was supported by Empirical Valence Bond (EVB) simulations, which revealed

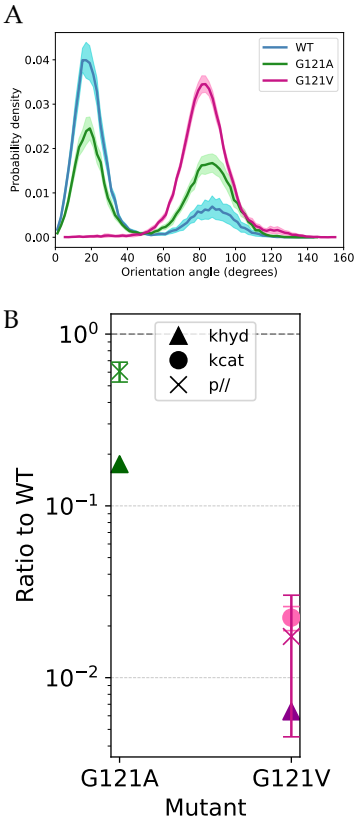

Figure 7.3: Conformational equilibrium in DHFR. (Figures made by Paul G.)
A. Distribution of the relative orientation between substrates in degrees for the last 150ns out of 400ns REMD simulation. For the 3 sequences tested, two different conformations of the substrates appear: one in which the angle is lower than 50°, that we call «parallel conformation», and one in which the angle is larger than 50° , that we call «perpendicular conformation».
B. Crosses indicate, for each mutant, the ratio relative to the WT of the probability to be in the parallel conformation. These values are compared to experimental measurements of k_{hyd} (triangles) from [164], and to a k_{cat} measurement (circle) for the G121V mutant obtained by Karolina F. and Kim R. in the context of this study. The color coding for the two mutants matches that used in Figure A.

4: More precisely, this concerns the orientation between the nicotinamide ring of the cofactor and the pteridine ring of the folate.

that the parallel conformation indeed corresponds to an active state, while the perpendicular conformation can be considered as inactive.

Based on all these observations, Paul G. proposed using the probability of adopting the parallel conformation as an indicator of enzymatic activity. As shown in Figure 7.3B, this metric, computed as a ratio relative to the WT, is consistent with experimental measurements of k_{cat} and k_{hyd}.

Building on this understanding obtained from MD simulations, compensatory mutations for G121V should aim to shift the equilibrium back toward the parallel (active) conformation, by reducing the steric hindrance between the Met20 and FG loops. This is precisely what we will explore in the following section using the methodology described in Chapter 6.

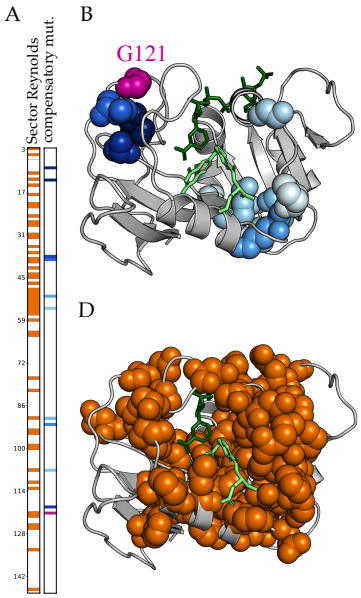

Figure 7.4: Compensatory mutations in the G121V mutant
A. The same mutations mapped onto a linear schematic based on the PDB 1RX2 sequence. Mutated residues are compared to those identified as part of the sector described by Reynolds *et al.* [36]. For compensatory mutations, the same blue gradient as in panel B is used; for sector residues, the gradient indicates the four levels of statistical confidence reported in the study.
B. Ribbon representation of *E. coli* DHFR (PDB: 1RX2) with the first 10 compensatory mutations shown as spheres colored from dark to light blue, indicating mutational order. G121 is highlighted in magenta.
C. Ribbon representation of *E. coli* DHFR (PDB: 1RX2) showing all sector residues (49 positions, orange) as defined by Reynolds *et al.* [36], covering approximately 33% of the sequence.

7.4 Compensating G121V using a statistical model

The methodology follows the approach described in Section 6.5, using an MSA made by Kalmer *et al.* [170]. This alignment includes 3664 sequences ($M_{\text{eff}} = 2664$) with 146 positions.

Once the SBM model is trained on this alignment, we obtain the fields and couplings needed to compute the statistical energy of any sequence. Specifically, we aim to identify successive mutations that minimize the following $\Delta\Delta E$ as we move away from the wild-type (WT) sequence:

$$\Delta\Delta E^{\text{G121V},\,d}_{\text{EcDHFR}}\left(\sigma_l : \gamma \to \beta\right) = \Delta E^{\text{V121},\,d}\left(\sigma_l : \gamma \to \beta\right) - \Delta E_{\text{EcDHFR}}\left(\sigma_l : \gamma \to \beta\right). \tag{7.1}$$

Here, G121V represents the mutation we aim to compensate, while EcDHFR denotes the *E. coli* DHFR sequence. For $d = 1$, we compute $\Delta E^{\text{G121V},\,1}\left(\sigma_l : \gamma \to \beta\right)$ across all positions l and possible substitutions β, and introduce the mutation yielding the most negative $\Delta\Delta E^{\text{G121V},\,1}$—which in this case is V13G. For $d = 2$, we repeat the same process, this time computing $\Delta E^{\text{G121V},\,2}\left(\sigma_l : \gamma \to \beta\right)$ based on the double mutant G121V V13G. By iteratively applying this procedure, we generate a series of mutations predicted to progressively compensate for preceding ones.

Before discussing the mutations suggested by this method, it is worth noting that only 28 sequences in the MSA contain a valine at position 121. These sequences share, on average, 35% sequence identity with *E. coli* DHFR, with the closest one reaching 44% identity to the wild-type. Predicting compensatory mutations for G121V from such highly divergent sequences may appear to contradict the importance of sequence context. However, as discussed in Chapter 6, the approach also relies on the sparsity of the epistatic interaction network to make such predictions possible [132].

The successive mutations predicted by the method are shown in Figure 7.4 and compared to the results of a sector analysis conducted by Reynolds *et al.* in 2011 on an MSA containing 418 sequences [36]. This sector contains 49 residus which represents more than 30% of the *E. coli* DHFR sequence. We observe that 6 out of the first 10 compensatory mutations belong to this sector. While this is more than expected on average if ten residues were chosen at random, it is not sufficient to reach statistical significance (p-value ~ 0.07)[5].

5: A useful follow-up analysis would be to compute compensatory mutations using the alignment from Reynolds *et al.*, and conversely, to perform the sector analysis using the alignment employed in the present study, in order to compare the results.

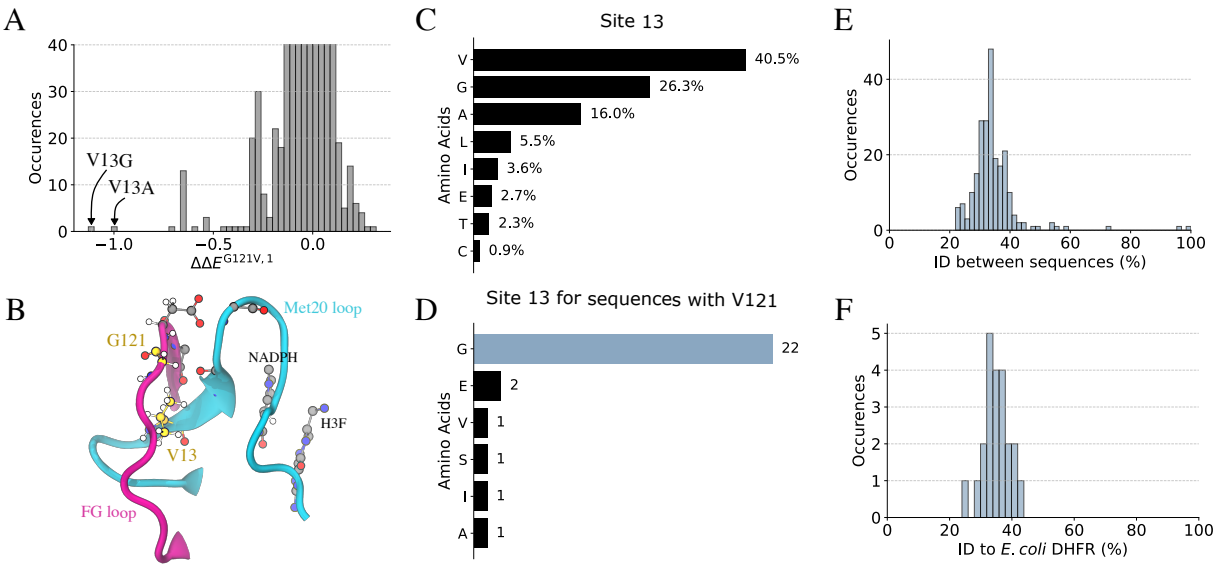

Figure 7.5: V13G as a compensatory mutation for G121V.
A. Distribution of $\Delta\Delta E$ ($\sigma_l : c \rightarrow d$) values across all positions and substitutions for the G121V mutant. Mutations V13G and V13A have respectively the most negative and second most negative $\Delta\Delta E$.
B. Zoom on the FG loop in pink and the Met20 loop in cyan of the *E. coli* DHFR structure obtained by Paul G. during a REMD simulation. Residu G121 and V13 are highlighted in yellow with their side chains. The nicotinamide ring of the cofactor and the pteridine ring of the folate are also displayed.
C. Amino acid distribution at site 13 in the alignment used to train the statistical model [170]. Only the 8 most frequent amino acids are shown.
D. Amino acid distribution at site 13 only for sequences containing a Valine (V) at site 121. Only 28 sequences feature a valine at site 121, and 22 of these also have a glycine at site 13.
E. Distribution of Identities between all pairs of the 22 sequences containing a Valine (V) at site 121 and a glycine (G) at site 13. These sequences originate from different organisms and share, on average, 34% sequence identity among themselves.
F. Distribution of Identities between the same 22 sequences and the wild-type sequence of *E. coli* DHFR. These sequences share, on average, 35% sequence identity with *E. coli* DHFR, with the closest one reaching 44% identity to the wild-type.

Recall that the steric hindrance introduced by the side chain of valine at site 121 plays a key role in confining the G121V mutant to the inactive perpendicular conformation. A natural strategy to relieve this constraint would be to mutate a residue on one of the two loops involved. Strikingly, the first mutation suggested by the method, V13G, targets a residue on the Met20 loop positioned directly opposite site 121. This particular mutation will therefore be the focus of the next section.

7.4.1 Focus on V13G mutation

In Figure 7.5A, $\Delta\Delta E_1^{G121V}$ ($\sigma_{13} : V \rightarrow G$) emerges as a clear outlier compared to the distribution of $\Delta\Delta E_1^{G121V}$ $(\sigma_l : \gamma \rightarrow \beta)$ values across all positions and substitutions. Notably, the second most strongly predicted compensatory mutation, V13A, also concerns residue 13. As shown in Figure 7.5B, this amino acid is located on the Met20 loop, opposite residu 121, which is on the FG loop.

A closer inspection of residue 13 in the alignment, shown in Figure 7.5C, reveals that valine is the most common amino acid at this position (40%), followed by glycine (26%) and alanine (16%). Based only on these frequencies, the V13G mutation would thus be predicted as deleterious, independent of any sequence context.

However, the statistical model also accounts for pairwise amino acid correlations, which are incorporated into the prediction of compensatory mutations through the statistical energy calculations. As illustrated in the

bar plot of Figure 7.5D, the model seems to have captured the following correlation signal: sequences containing a valine (V) at position 121 almost all have a glycine (G) at position 13.

Although the V121 G13 combination is observed in only 22 sequences across the alignment, these sequences originate from diverse organisms and share, on average, 34% sequence identity among themselves (Figure 7.5E). While weighted statistics are employed to mitigate sampling biases within the MSA, it is interesting to emphasize that the model's prediction is here driven by a correlation signal supported by substantial sequence diversity.

This observation is even more important given that the alignment does not contain any close homologs of the wild-type sequence featuring a valine at site 121. The closest sequence exhibiting the V121 G13 combination comes from an other bacterial species (Geomicrobium sp. JCM 19039) and shares only 44% identity with *E. coli* DHFR. It is thus important to note that these V121 G13 sequences exhibit similar levels of divergence from each other as from *E. coli* DHFR.

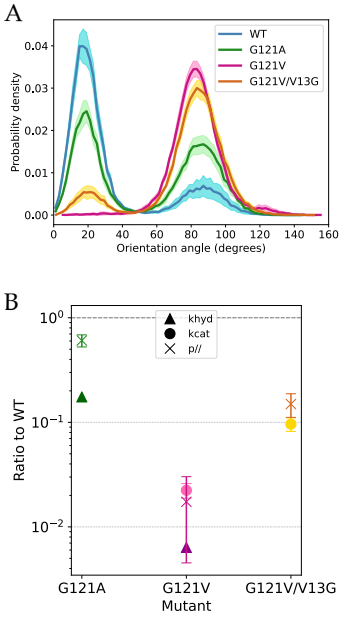

Figure 7.6: Validation of the compensatory mutation V13G. (Figures made by Paul G.)
A. Distribution of the relative orientation between substrates in degrees for the last 150ns out of 400ns REMD simulation. We show the same distributions as in Figure 7.3, and we add the distribution for the double mutant G121V V13G in orange. The latter exhibit a partial shift of the equilibrium towards the parallel conformation.
B. Crosses indicate, for each mutant, the ratio relative to the WT of the probability to be in the parallel conformation. These values are compared to experimental measurements of k_{hyd} (triangles) from [164], and to k_{cat} measurements (circle) for the G121V and G121V V13G mutants obtained by Karolina F. and Kim R. in the context of this study. The color coding for the two mutants matches that used in Figure A.

7.5 In Silico & Experimental Validation

To evaluate the capacity of the V13G mutation to compensate for the deleterious effect of G121V on the enzymatic activity, Paul G. explored the conformational space of the G121V V13G double mutant. As illustrated in Figure 7.6, MD simulations revealed a partial shift of the equilibrium toward the parallel (active) conformation, corresponding to a tenfold increase in the probability of occupying this state.

Figure 7.7 presents a structural comparison of the Met20 and FG loops based on representative conformations from REMD simulations of the wild-type, G121V, and G121V V13G sequences. For the double mutant, the parallel conformation illustrated in Figure 7.7C closely mirrors the wild-type structure shown in Figure 7.7A. The G121V mutant, in contrast, is distinguished by the presence of two valine residues with extended side chains at sites 121 and 13, creating a steric hindrance that separates the two loops. Substitution of the valine at position 13 with a glycine, which has a smaller side chain, can reduce the steric hindrance, thereby facilitating the formation of parallel conformations.

In order to confirm these predictions, Karolina Filipowska and Kim Reynolds quantified the catalytic activity of the wild-type enzyme, as well as the G121V and G121V V13G mutants. As depicted in Figure 7.6, the experimental measurements are consistent with the computational results. Notably, the introduction of the V13G mutation was found to enhance the k_{cat} by a factor of 2.8 relative to the G121V mutant.

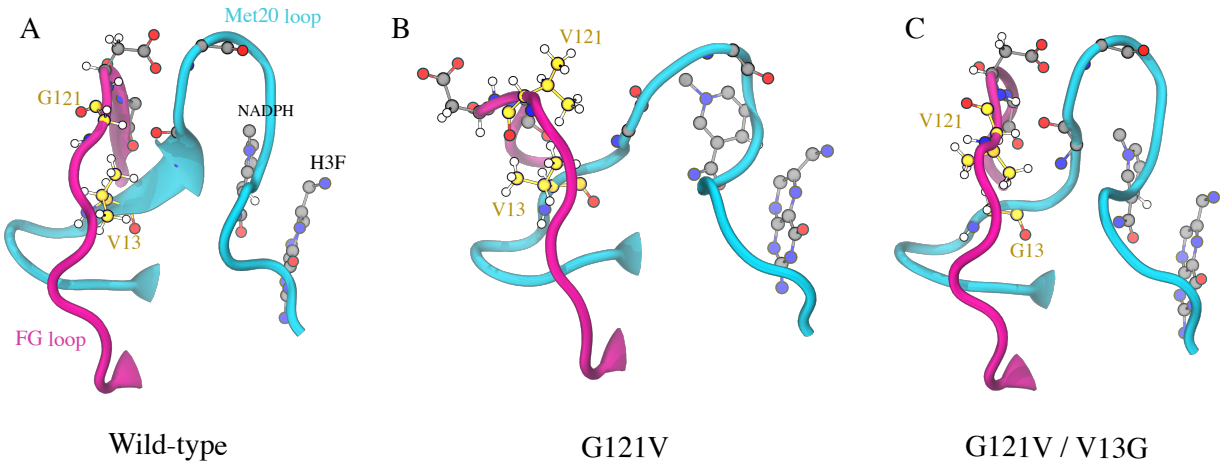

Figure 7.7: Zoom on the DHFR structure. (Figures made by Paul G.)
The three figures show a zoom on the FG loop in pink and the Met20 loop in cyan for 3 different sequences. The two residus of interest 121 and 13 are highlighted in yellow. In the background, we can also observe the relative orientation of the nicotinamide ring of the cofactor NADPH and the pteridine ring of the folate H_3F.
A. Example of a parallel conformation obtained with the Wild-type *E. coli* DHFR sequence. The glycine (G) at site 121 has a short side chain while the Valine (V) at site 13 has a long side chain.
B. For the mutant G121V, the Glycine at site 121 is replaced by a Valine that has a longer side chain. This creates a steric hindrance that pushes the two loops apart. This is an example of an inactive conformation in which the orientation between the cofactor and the substrate is perpendicular.
C. For the double mutant G121V / V13G, the long side chain of the Valine at site 13 is replaced by the short side chain of the Glycine. This can help to reduce the steric hindrance and increase the chance of this sequence to be in the parallel conformation displayed here.

7.6 Conclusions

In this chapter, we focused on the enzyme dihydrofolate reductase (DHFR), a widely studied system known for its allosteric properties.

Using the COMBINE method introduced in Chapter 6, we identified potential compensatory mutations for the G121V mutant, a deleterious substitution that disrupts DHFR catalytic activity despite being distant from the active site.

We compared the mutations proposed by COMBINE to the allosteric sector identified by Reynolds *et al.* While many of the predicted mutations fall within or near this sector, further analysis would be needed to rigorously compare the results of these two approaches.

Interestingly, the first mutation proposed by COMBINE targets residue 13, located on the Met20 loop, directly across from position 121. This prediction aligns well with molecular dynamics simulations performed by Paul Guenon, which indicated that the G121V mutation induces steric hindrance between the FG and Met20 loops. Replacing valine with glycine at position 13—thus shortening the side chain—was expected to relieve this steric hindrance.

This hypothesis was tested both in silico and experimentally. MD simulations made by Paul Guenon revealed a partial restoration of the active conformation, and experimental measurements by Karolina Filipowska and Kim Reynolds confirmed that the V13G mutation increases catalytic efficiency relative to the G121V mutant.

Although the rescue is only partial, the data-driven predictions made with COMBINE clearly supports the understanding of the allosteric mechanism obtained through MD simulations.

More broadly, this study highlights the value of integrating complementary approaches to investigate protein properties. Here, we combined physics-based molecular simulations, predictions informed by evolutionary data, and experimental measurements.

8 Conclusions

As we reach the end of this manuscript, it is worth revisiting the questions that motivated this work and summarizing the insights gained along the way.

At the heart of this project are proteins and the use of statistical inference as a tool to study their properties. We have focused in particular on Boltzmann Machine models, which have been widely studied and applied in the context of protein sequences.

The first part of this work was motivated by two key observations reported in the literature: (i) Boltzmann Machines trained on a protein family can generate functional sequences within that family. However, because of the undersampled regime, regularization becomes necessary, introducing biases such that generating functional sequences requires rescaling the model [58]; (ii) in this same regime, the inferred Potts model does not reliably capture interactions across different scales. [30].

In Chapter 3, we revisited these important results from the literature and showed, in particular, that lowering the sampling temperature leads to a loss in both the novelty and diversity of the generated sequences.

Building on this, we chose to continue working within the Potts model framework and focused on adapting the inference procedure, hypothesizing that another optimization strategy could more faithfully capture the structure of the data.

This direction is developed in Chapter 4, where we introduced the SBM (Stochastic Boltzmann Machine) inference method. We demonstrated that, on a toy model, SBM inference more accurately recovers interactions occurring at different scales in the data.

Encouraged by these results, we applied the method to real data, focusing on the chorismate mutase family. Two key findings emerged: (i) The SBM model can generate functional sequences at T=1, thus preserving the diversity of natural sequences; (ii) Increasing regularization to produce more divergent sequences leads to a loss in generative capacity, even though the sequences remain statistically indistinguishable from natural ones according to our in silico analyses.[1]

1: Additional experimental validation is underway using intermediate levels of regularization to test the model's generalization capacity.

These results call for a theoretical understanding of the SBM method, but they also raise several questions: Is its apparent advantage over standard Boltzmann Machines on real data rooted in its ability to capture multiscale interactions, as suggested by the toy model? What is the fundamental limit of novelty that can be reached with a given dataset and model? Beyond reproducing the distribution of natural sequences, can these models be used to investigate specific properties of proteins?

This last question motivated the second major part of this thesis.

In this context, we focused on serine proteases, a family of enzymes widely studied for their functional diversity and already under investigation by close collaborators. Like many enzymes, they exhibit strong substrate specificity, cleaving peptides only after particular amino acids. Yet, the mechanisms underlying this specificity remain only partially understood.

As in the undersampling problem, several key results from the literature shaped our approach: (i) The binding pocket contains residues that are critical for specificity, notably residue 189, which is in direct contact with the substrate; (ii) Previous attempts to switch specificity by mutating residue 189 have failed. For the trypsin-to-chymotrypsin conversion, the only successful attempt required 16 mutations, most of which do not directly contact the substrate [143]; (iii) The mutations involved in the trypsin-to-chymotrypsin conversion align with a sector identified by SCA and experimentally validated as contributing to substrate specificity [19, 29].

In Chapter 6, we used the prediction of compensatory mutations, triggered by a mutation at residue 189, as a strategy to identify epistatic interactions relevant to specificity.

Using this approach, we showed that a chymotrypsin-like activity, though partial, could be achieved with only three mutations located in the binding pocket of rat trypsin. This suggests that specificity can be significantly modulated through a limited number of mutations confined to the binding pocket. More generally, this result further illustrates the ability of the Boltzmann Machine model to identify epistatic interactions from sequence data.

However, in the reverse direction, from chymotrypsin to trypsin, none of the tested mutants displayed measurable trypsin activity. This mirrors previous observations and raises the question of why acquiring trypsin-like specificity on a chymotrypsin scaffold appears to be a more challenging task.

In Chapter 7, we extended our approach to a different system: dihydrofolate reductase, which is well known for its allosteric properties. This work centered on a mutation distant from the active site that is known to drastically reduce enzymatic activity.

The mechanism of allostery, although not detailed in this manuscript, was investigated through molecular dynamics simulations performed by Paul Guenon. Interestingly, a compensatory mutation predicted independently by our statistical model was found to be consistent with the allosteric mechanism inferred from these simulations. This observation motivated an experimental validation, which confirmed that the predicted mutation could partially restore enzymatic activity. While it confirmed the predictive power of the statistical model, this result most importantly provided additional support for the proposed allosteric mechanism.

To close this manuscript, I would like to leave the reader with two final remarks.

First, throughout this work, we have often emphasized the exponential growth of biological data, which has enabled the development of new lines of research. In this context, there has been a growing tendency to assemble large datasets, use deep learning models, and focus on predictive accuracy. While there is little doubt that modern machine learning will remain a key tool for analyzing high-dimensional biological data, I am also convinced that these models are, at best, a starting point for understanding, not its endpoint.

Second, what has truly given depth to the projects presented here is the opportunity to collaborate, on each of them, with students and researchers working on proteins through complementary approaches.

These collaborations have not only enriched this scientific work but have also made this journey intellectually and personally rewarding.

Appendix

Technical details on Boltzmann Machine models

A

A.1 Gauge invariance

The pairwise potts model presented in Section 2.1 remains unchanged under the tranformation [43]:

$$\begin{aligned} J_{ij}(a,b) &\leftarrow J_{ij}(a,b) - K_{ij}(a) + K_{ji}(b) \\ h_i(a) &\leftarrow h_i(a) - g_i + \sum_{j\neq i} \left[K_{ij}(a) + K_{ji}(a)\right] \end{aligned}$$

where $K_{ij}(a)$ and g_i are arbitrary quantities.

In the case of the *zero-sum-gauge*, this transformation can be explicitly written as:

$$J_{ij}(a,b) \leftarrow J_{ij}(a,b) - \frac{1}{q}\sum_{b=1}^{q} J_{ij}(a,b) - \frac{1}{q}\sum_{a=1}^{q} J_{ij}(a,b) + \frac{1}{q^2}\sum_{a,b=1}^{q} J_{ij}(a,b),$$

$$h_i(a) \leftarrow h_i(a) - \frac{1}{q}\sum_{b=1}^{q} h_i(b) + \sum_{j\neq i}\left[\frac{1}{q}\sum_{b=1}^{q} J_{ij}(a,b) - \frac{1}{q^2}\sum_{a,b=1}^{q} J_{ij}(a,b)\right]$$

Note that there exist other gauge transformation commonly used in the literature, such as the *lattice-gas gauge*, in which $h_i(q) = J_{ij}(a,q) = J_{ij}(q,a)\ \forall i,j,a$ so that energies are measured with respect to the configuration $(q,...q)$.

A.2 Metropolis-Hasting algorithm

The MLE framework presented in Section 2.1.2 requires sampling sequences from the Boltzmann distribution defined in Equation 2.2. This can be achieved using the Metropolis-Hastings algorithm [50, 51], which we briefly describe below.

Given a current sequence $\boldsymbol{\sigma}^{(k)}$, the algorithm proceeds as follows:

1. Propose a new sequence $\boldsymbol{\sigma}'$ by randomly selecting a site $i \in \{1,\dots,L\}$ and replacing σ_i with a new amino acid $a' \in \{0,\dots,20\}$, chosen uniformly at random (excluding the current amino acid).
2. Compute the energy difference $\Delta E = E(\boldsymbol{\sigma}') - E(\boldsymbol{\sigma})$, using the energy function defined in Equation 2.3.
3. Accept the proposed move with probability $\min\left(1, e^{-\Delta E}\right)$.
4. If the move is accepted, set $\boldsymbol{\sigma}^{(k+1)} = \boldsymbol{\sigma}'$; otherwise, retain the current state: $\boldsymbol{\sigma}^{(k+1)} = \boldsymbol{\sigma}^{(k)}$.

Iterating this procedure for a sufficiently large number of steps (k_{mc}) yields a synthetic sequence approximately sampled from the target distribution $P(\boldsymbol{\sigma} \mid \boldsymbol{\theta})$.

To obtain N_{chains} sequences, one option is to run a single chain and store sequences separated by a sufficient number of steps, ensuring they are decorrelated. In this work, however, we adopt the following strategy:

N_{chains} independent Markov chains are run in parallel, and the sequences obtained after k_{mc} steps are retained as samples.

Note that the same algorithm is also used to generate synthetic sequences after the training procedure is complete.

A.3 Non-equilibrium regime in the learning of BM for protein sequences

We provide here additionnal results, studying the diversity of artificial datasets generated with a BM trained on an MSA of the Chorismate Mutase family (length L=96, M=1258 sequences). The models were obtained using a the MLE framework with a Vanilla gradient descent and the MCMC simulations were performed starting from random sequences, with $k_{\text{mc}} = 10^5$ and $N_{\text{chains}} = 1000$.

In Figure A.1, we observe that when too few MCMC steps are used during inference, the diversity of the generated sequences becomes highly sensitive to the number of sampling steps. In particular, the longer the sampling runs, the more diversity is lost.

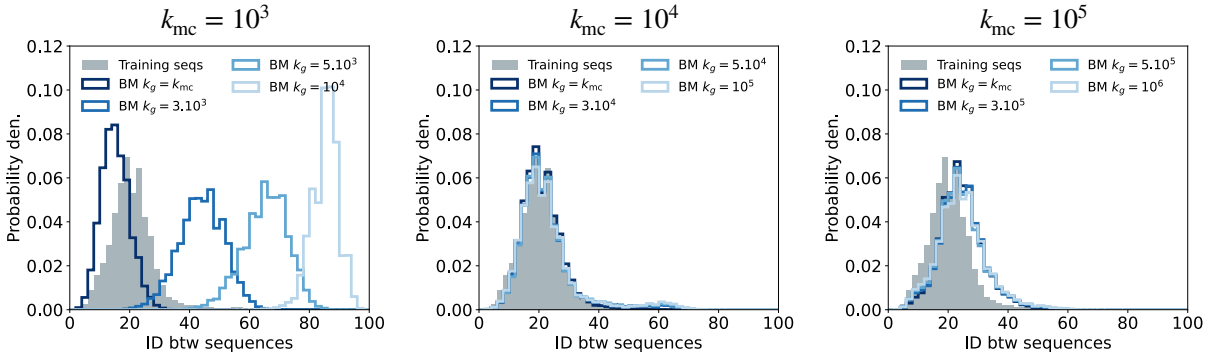

Figure A.1: Diversity of synthetic datasets as a function MCMC steps
The three panels display histograms of sequence identity between pairs of sequences within natural or synthetic datasets. Synthetic sequences were generated using different numbers of MCMC steps k_g, chosen equal to or greater than the number of MCMC steps k_{mc} used during training. From left to right: $k_{\text{mc}} = 10^5$, $k_{\text{mc}} = 10^4$, and $k_{\text{mc}} = 10^3$.

Supplementary Undersampling biases

B

B.1 Sampling at T<1

We here provide additional results illustrating the effect of the sampling temperature on the novelty and diversity of artificial sequences.

B.1.1 Real data

We investigate the impact of sampling temperature on the diversity and novelty of artificial sequences generated in the study by Russ *et al.* [58]. For conciseness, only a subset of these results is presented in the main text. However, similar trends—namely a loss of novelty and diversity at low temperatures—are also observed for the model inferred with regularization strength $\lambda = 0.0001$, and are shown here for completeness (see Figure B.1).

In Figure B.2 and Figure B.3, we present the full bar plots of family representation for the two regularization strengths and the three sampling temperatures. Notably, not all families are represented in the artificial data even at $T = 1$. However, this observation should be interpreted with caution, as it also reflects the fact that the number of generated sequences is smaller than the number of families in the natural dataset. In Figure C.6, we report similar statistics computed over 10,000 sequences sampled from a BM model at $T = 1$, showing that 93.8% of the natural families are represented.

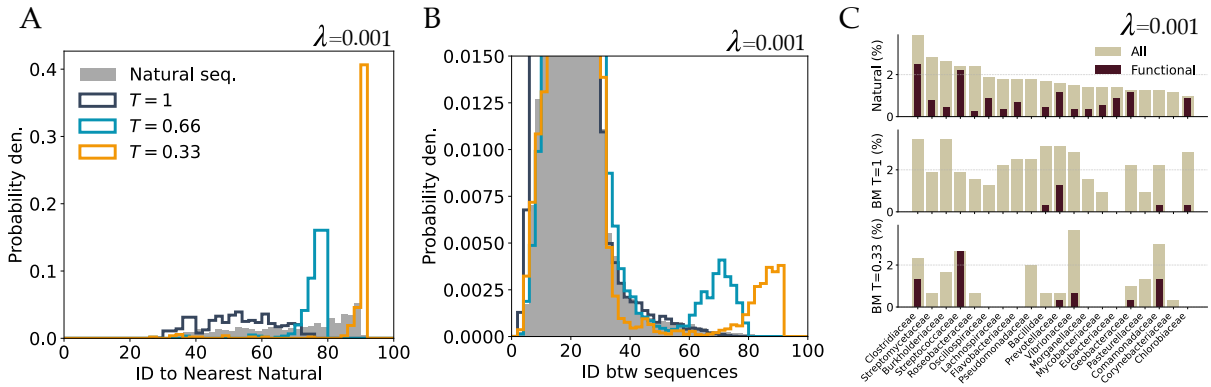

Figure B.1: CM family: Artificial sequences as a function of sampling temperature for $\lambda = 0.001$. (Figures based on data from Russ *et al.* [58])
A. Histogram of sequence identity (ID) to the nearest natural sequence, for three sampling temperatures ($T = \{1, 0.66, 0.33\}$).
B. Histogram of pairwise identity percentages among natural sequences and artificial sequences generated at the same three temperatures. As the sampling temperature decreases, the distribution diverges from that of the natural sequences, with a growing proportion of sequence pairs exhibiting identity values above 60%.
C. Top bar plot: proportion of natural sequences belonging to the 20 most represented families in the natural alignment. For comparison, analogous statistics are shown for artificial sequences generated using a BM at $T = 1$ and $T = 0.33$; in these cases, each artificial sequence is assigned to the family of its closest natural counterpart. Notably, lowering the sampling temperature introduces a bias toward certain families. The high proportion of functional sequences observed at $T = 0.33$ may be attributed to this effect.

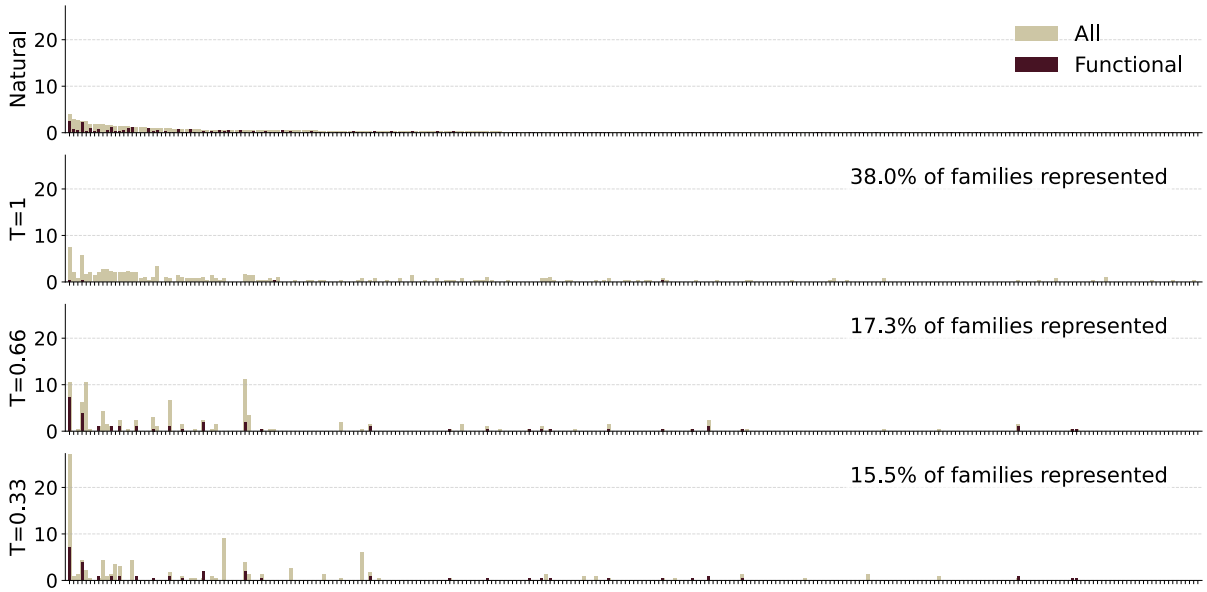

Figure B.2: CM family: Effect of sampling temperature on family diversity of artificial sequences for $\lambda = 0.01$. (Figures based on data from Russ *et al.* [58])
Each bar represents the number of sequences assigned to one of the 271 protein families identified in the natural dataset. For synthetic sequences, assignments are based on the closest natural sequence in terms of identity. Natural sequences are shown in the top panel for reference. Lower panels display the distribution of family representation for artificial sequences generated at three sampling temperatures: $T = 1$, $T = 0.66$, and $T = 0.33$. As the temperature decreases, a clear bias emerges toward a smaller subset of families, with only 38.0%, 17.3%, and 15.5% of the families represented at $T = 1$, $T = 0.66$, and $T = 0.33$, respectively. Bars in light beige represent all artificial sequences; overlaid dark bars indicate sequences experimentally labeled as functional.

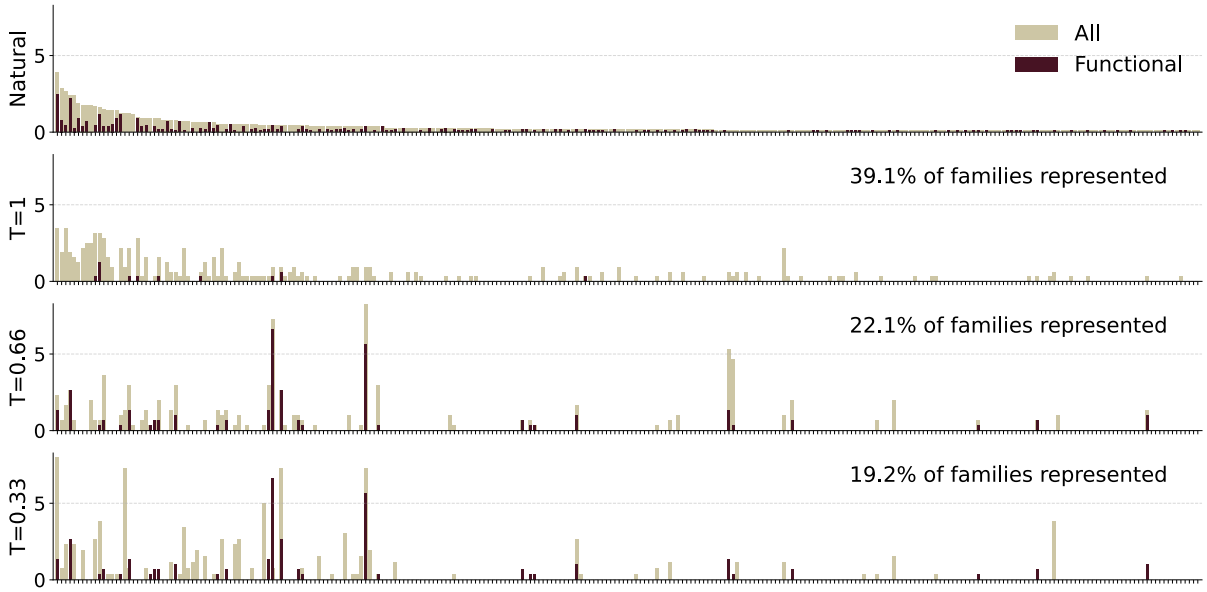

Figure B.3: CM family: Effect of sampling temperature on family diversity of artificial sequences for $\lambda = 0.001$. (Figures based on data from Russ *et al.* [58])
Each bar represents the number of sequences assigned to one of the 271 protein families identified in the natural dataset. For synthetic sequences, assignments are based on the closest natural sequence in terms of identity. Natural sequences are shown in the top panel for reference. Lower panels display the distribution of family representation for artificial sequences generated at three sampling temperatures: $T = 1$, $T = 0.66$, and $T = 0.33$. As the temperature decreases, a clear bias emerges toward a smaller subset of families, with only 38.0%, 17.3%, and 15.5% of the families represented at $T = 1$, $T = 0.66$, and $T = 0.33$, respectively. Bars in light beige represent all artificial sequences; overlaid dark bars indicate sequences experimentally labeled as functional.

B.1.2 Toy Model

In this section, we present additional results illustrating the impact of low-temperature sampling on the diversity of generated sequences in the context of a toy model.

The setup is identical to that described in Section 3.3.1 of Chapter 3. Sequences were sampled at three different temperatures, and we show that, consistent with observations on real data, lowering the temperature leads to a marked reduction in both diversity and novelty.

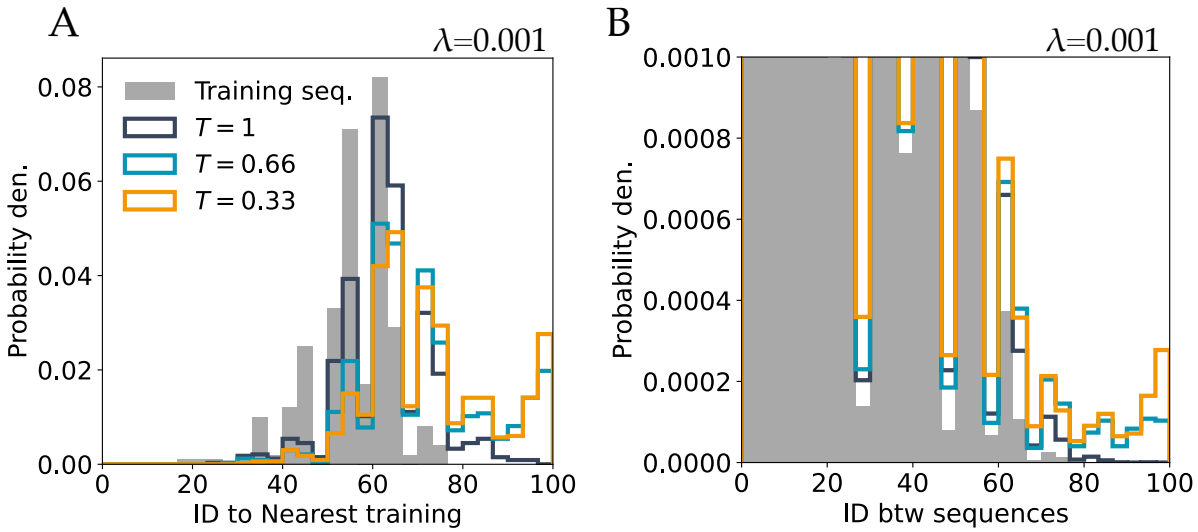

Figure B.4: Toy model: effect of sampling temperature on diversity and novelty.
A. Histograms of sequence identity (ID) to the nearest training sequence, for artificial sequences generated at three different temperatures ($T = \{1, 0.66, 0.33\}$), compared to the natural dataset.
B. Histograms of pairwise sequence identity among training sequences and artificial sequences generated at the same three temperatures. As the sampling temperature decreases, the distribution progressively deviates from that of the training set, with a growing number of sequence pairs exhibiting identity values above 60%.

Supplementary Stochastic Boltzmann Machine

C

C.1 Algorithm

We provide here additional technical details on the implementation of the SBM method, which relies on the L-BFGS optimization algorithm.

The core idea of L-BFGS is to incorporate curvature information from recent iterations in order to build an approximation of the inverse Hessian. In our implementation, we use a simple and efficient procedure to compute the search direction $\boldsymbol{p}_t = -\boldsymbol{H}_t \nabla f(\boldsymbol{\theta}_t)$, based on the m most recent pairs of vectors $\{\boldsymbol{s}_t, \boldsymbol{y}_t\}$ stored during the optimization [113]. The algorithm used to compute this product is provided below:

Algorithm 2: Calculation of $\boldsymbol{p}_t = -\boldsymbol{H}_t \nabla f(\boldsymbol{\theta}_t)$

Input: $\boldsymbol{\theta}_t, \nabla f(\boldsymbol{\theta}_t), \{\boldsymbol{s}_i, \boldsymbol{y}_i\}_{i=t-m}^{t-1}$

$\boldsymbol{H}_t^0 = \frac{\boldsymbol{s}_{t-1}^T \boldsymbol{y}_{t-1}}{\boldsymbol{y}_{t-1}^T \boldsymbol{y}_{t-1}} I$;
$h = -\nabla f(\boldsymbol{\theta}_t)$;
for $i = t-1, t-2, \ldots, t-m$: **do**
  $\alpha_i = \frac{\boldsymbol{s}_i^T h}{\boldsymbol{y}_i^T \boldsymbol{s}_i}$;
  $h = h - \alpha_i \boldsymbol{y}_i$;
end
$h = \boldsymbol{H}_t^0 h$;
for $i = t-m, t-m+1, \ldots, t-1$: **do**
  $\beta = \frac{\boldsymbol{y}_i^T h}{\boldsymbol{y}_i^T \boldsymbol{s}_i}$;
  $h = h - \boldsymbol{s}_i(\alpha_i - \beta)$;
end
$\boldsymbol{p}_t = h$;

In general, learning rates must be chosen using a line search procedure to ensure that parameter updates satisfy the Wolfe conditions. However, this requires multiple evaluations of the objective function, resulting in a computational cost that is prohibitive in our context.

In our implementation, the initial learning rate is set to $\eta_0 = \frac{1}{\|\nabla f(\boldsymbol{\theta}_0)\|}$, and the first search direction is taken as $\boldsymbol{p}_0 = -\nabla f(\boldsymbol{\theta}_0)$. For all subsequent iterations, Algorithm 1 is applied and the learning rate η_t is fixed to 1.

If, at any iteration, the condition $\boldsymbol{y}_t^T \boldsymbol{s}_t < 0$ or $\boldsymbol{y}_t^T \boldsymbol{s}_t \approx 0$ is met, the inverse Hessian estimate is not updated and we set $H_{t+1} = H_t$. However, in some cases this condition may occur too frequently, preventing the estimate from properly capturing curvature information. This is a potential issue that we monitor carefully throughout the learning procedure.

Note that in the special case where $m = 1$, the inverse Hessian approximation is computed as:

$$H_t = V_{t-1}^T H_t^0 V_{t-1} + \frac{\boldsymbol{s}_{t-1} \boldsymbol{s}_{t-1}^T}{\boldsymbol{y}_{t-1}^T \boldsymbol{s}_{t-1}} \tag{C.1}$$

where $V_t = I - \frac{y_t s_t^T}{y_t^T s_t}$. This corresponds to the standard BFGS update formula, with the distinction that H_t^0 is used instead of H_t.

C.2 Learning dynamics of coupling features in the toy model

In this section, we provide supplementary figures illustrating the learning dynamics of the different types of couplings, across a range of hyperparameter settings. For each plot, the magnitudes of the inferred couplings are computed as averages—over each coupling type—of the Frobenius norm of the corresponding coupling matrices.

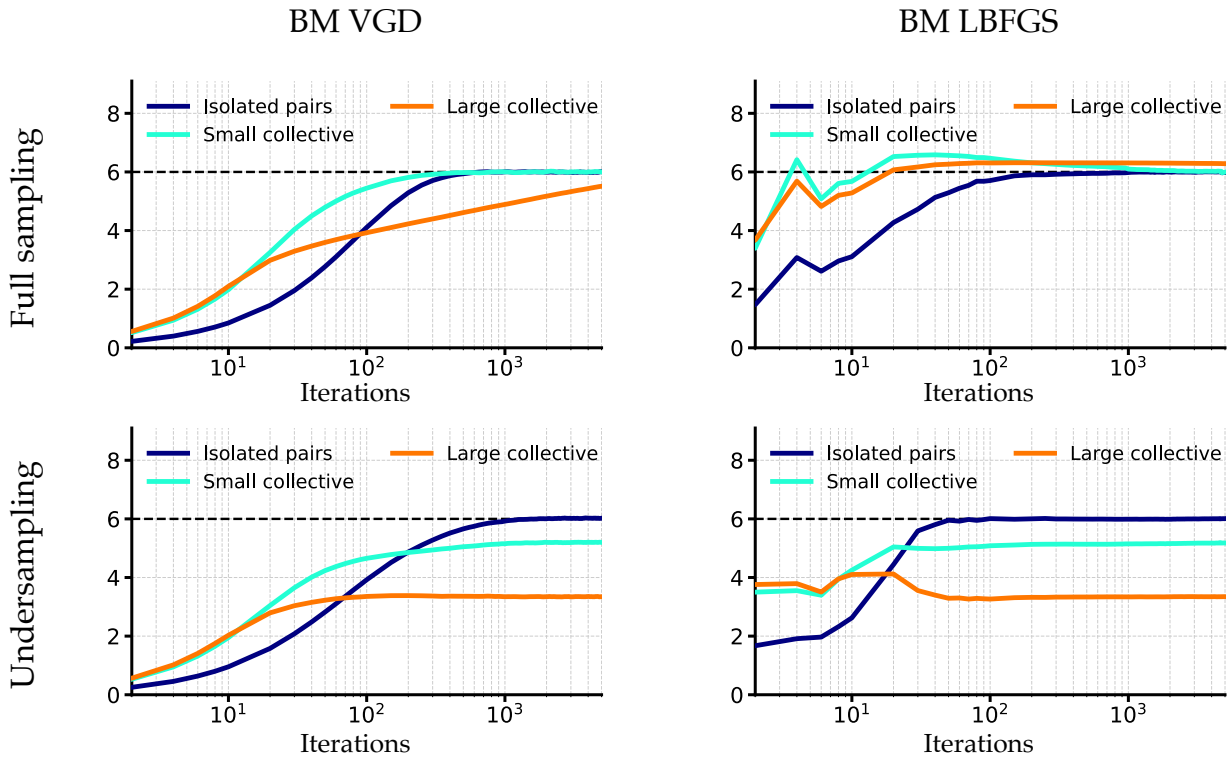

Figure C.1: Learning dynamics of coupling features in the toy model.
Evolution of the magnitude of inferred couplings as a function of the number of iterations, for four inference settings: a BM^{VGD} trained under full sampling (upper left), a BM^{VGD} under undersampling (lower left), a BM^{LBFGS} under full sampling (upper right), and a BM^{LBFGS} under undersampling (lower right). In each panel, the true coupling strength of the toy model is indicated by a dotted black line. Three curves are shown in each plot, corresponding to the three types of couplings: isolated pairs (blue), small collective features (green), and large collective features (orange). The undersampling regime is the same as in the main text, with $M = 300$, while the full sampling regime corresponds to $M = 100{,}000$ sequences.

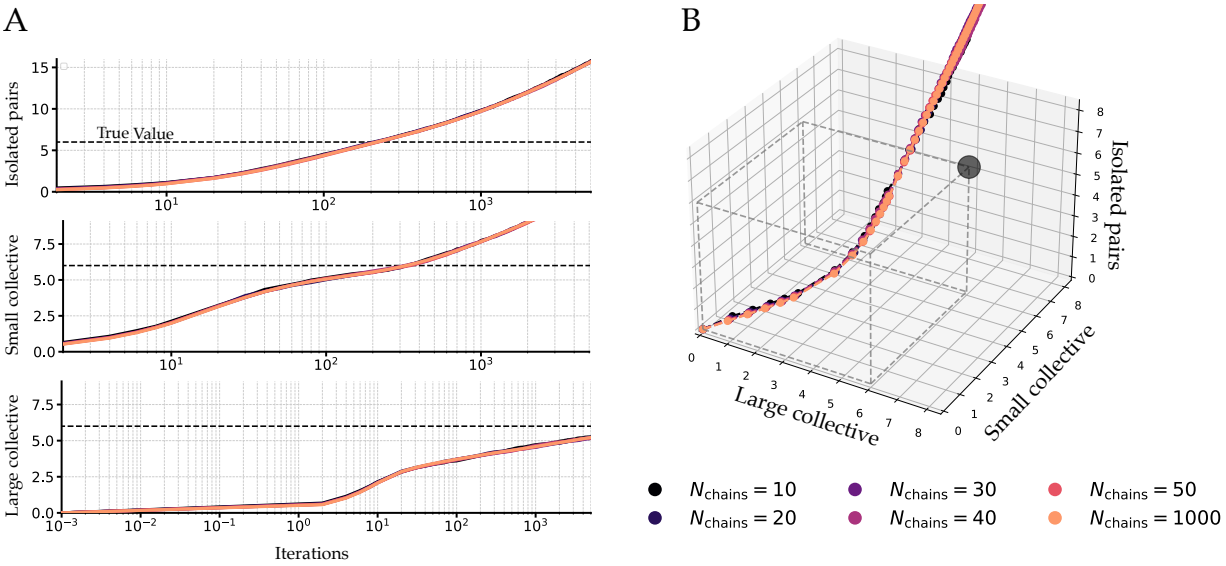

Figure C.2: Inference of toy model couplings as a function of N_{chains} for the BM$^{\text{VGD}}$ method.
For all these pannel, the magnitude of the couplings are computed from the frobenius norm of the coupling matrix, by averaging over the different types of interacting couplings. The models where inferred with the BM$^{\text{VGD}}$ method without L_2 regulararization.
A. Magnitude of the inferred couplings over iterations for different values of the hyperparameter N_{chains}. The true values of the toy model couplings are indicated by a dotted black line. The three panels correspond to the three types of couplings: isolated pairs (top), small collective groups (middle), and large collective groups (bottom).
B. Using the same data as in panel A, we show the inference trajectories in a 3D space defined by the isolated, small collective, and large collective couplings. Larger markers indicate that successive points in the trajectory are close in parameter space, reflecting a slowdown in the inference dynamics. The grey dotted cube indicates the target values in each direction, and the grey dot corresponds to the point where all interacting couplings are correctly inferred.

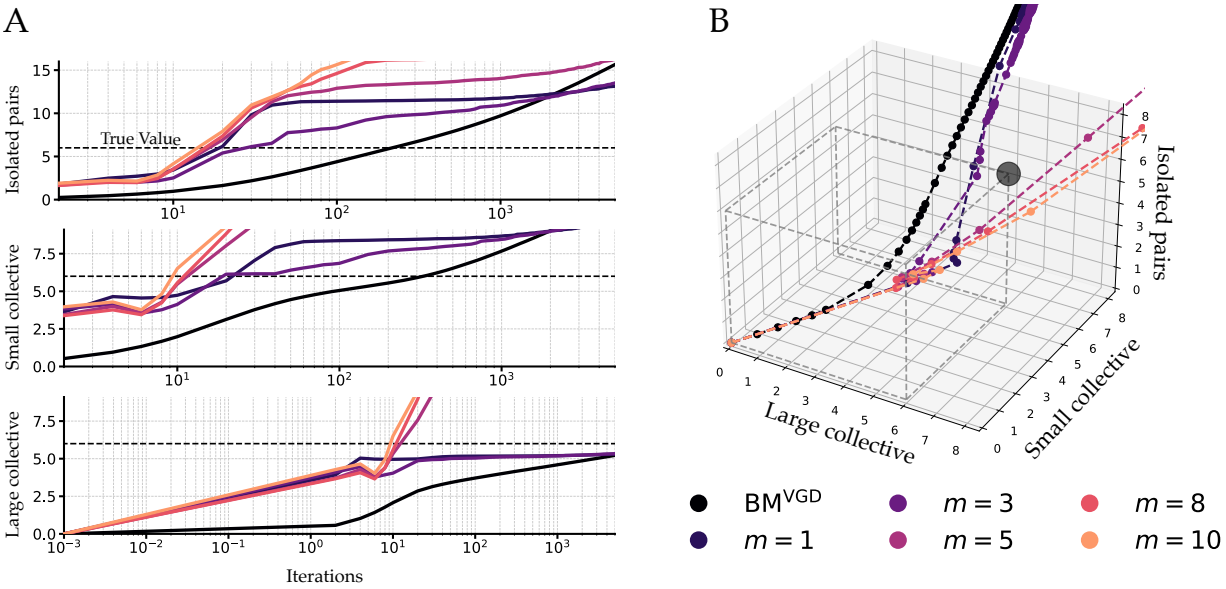

Figure C.3: Inference of toy model couplings as a function of m for the BM$^{\text{LBFGS}}$ method.
For all these pannel, the magnitude of the couplings are computed from the frobenius norm of the coupling matrix, by averaging over the different types of interacting couplings. $N_{\text{chains}} = 1000$
A. Magnitude of the inferred couplings over iterations for different values of the hyperparameter m. For comparison we also show the trajectory for BM$^{\text{VGD}}$ without regulararization. The true values of the toy model couplings are indicated by a dotted black line. The three panels correspond to the three types of couplings: isolated pairs (top), small collective groups (middle), and large collective groups (bottom).
B. Using the same data as in panel A, we show the inference trajectories in a 3D space defined by the isolated, small collective, and large collective couplings. Larger markers indicate that successive points in the trajectory are close in parameter space, reflecting a slowdown in the inference dynamics. The grey dotted cube indicates the target values in each direction, and the grey dot corresponds to the point where all interacting couplings are correctly inferred.

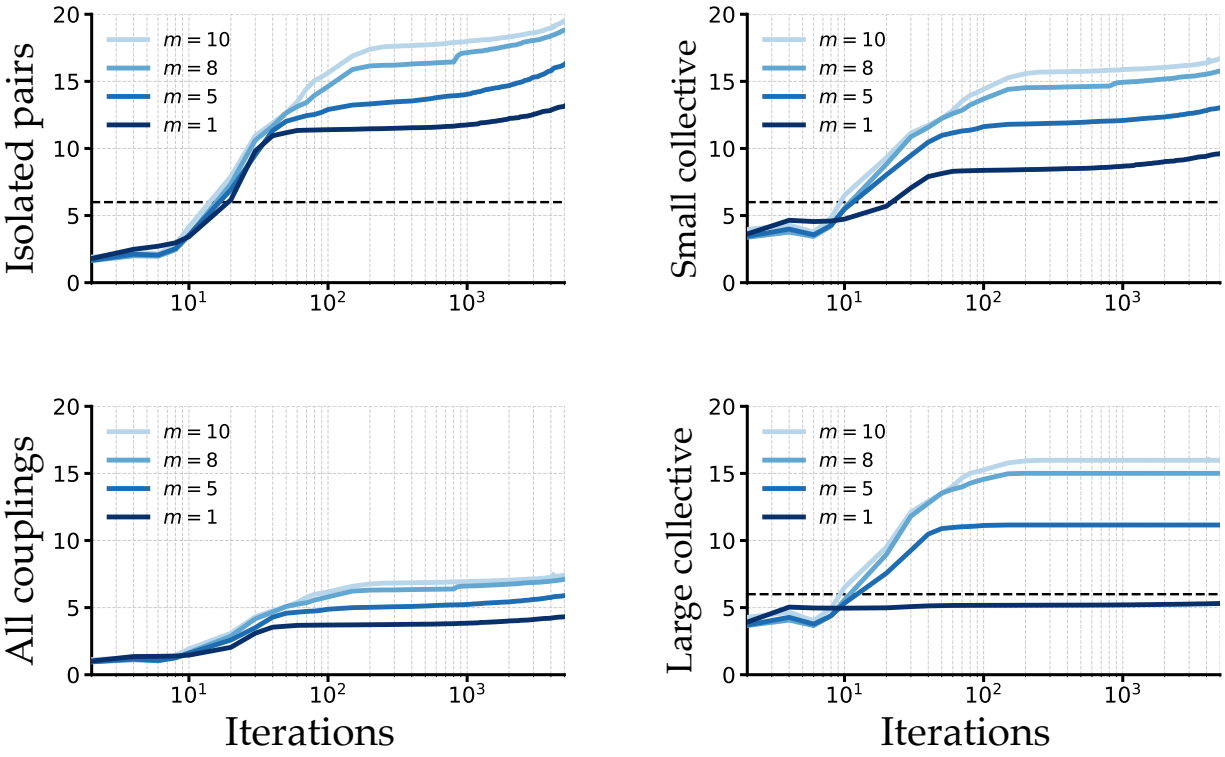

Figure C.4: Inference of toy model couplings as a function of m for the $\mathrm{BM}^{\mathrm{LBFGS}}$ method.
The magnitudes of the inferred couplings are computed by first taking the Frobenius norm of each coupling matrix, then averaging over each type of interaction. Inference is performed using the L-BFGS algorithm with varying values of the memory parameter m and no L_2 regularization. The plots show the coupling magnitudes as a function of training iterations for: isolated pairs (upper left), all couplings combined (lower left), small collective features (upper right), and large collective features (lower right). In each panel, the dynamics consistently exhibit a clear plateau, whose value decreases as m decreases.

C.3 Comparison of BM and SBM methods on toy model

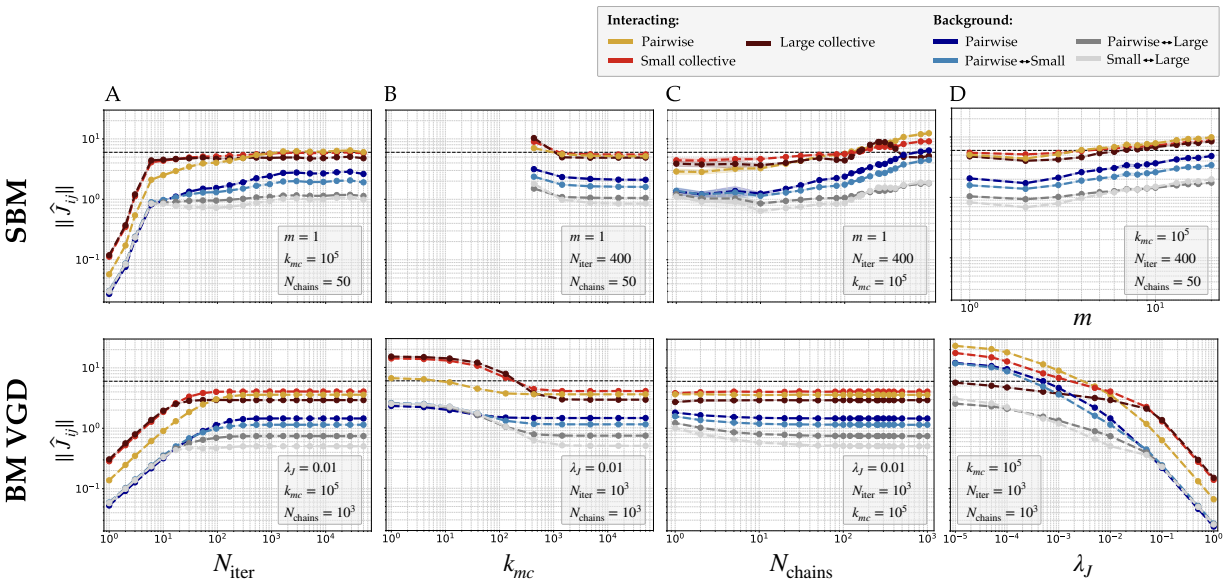

Figure C.5: Comparison of BM and SBM methods on the Toy Model across hyperparameter choices.
We evaluate the impact of various hyperparameters on the magnitude of inferred couplings $\|\hat{J}_{ij}\|$ for the different types of interacting couplings in the toy model. The first row corresponds to the SBM method, while the second row shows results for the BM method trained with vanilla gradient descent (VGD). Coupling magnitudes are computed as Frobenius norms over amino acids of the coupling matrices, averaged over couplings belonging to the same category. The true coupling amplitude is shown as a dashed black line. Colors distinguish between: true interacting couplings (pairwise: yellow, small collective: red, large collective: dark red) and background couplings (non-interacting) of different types (see legend). Each column explores a different hyperparameter:
A. Number of training iterations N_{iter}.
B. Number of MCMC steps k_{mc}.
C. Number of independent MCMC chains N_{chains}.
D. Regularization strength λ_J for BM. For SBM, panel **D** instead varies the memory parameter m used in L-BFGS.

C.4 Comparison of BM ($\lambda = 0.01$) and SBM ($N_{\text{chains}} = 50$) on CM family

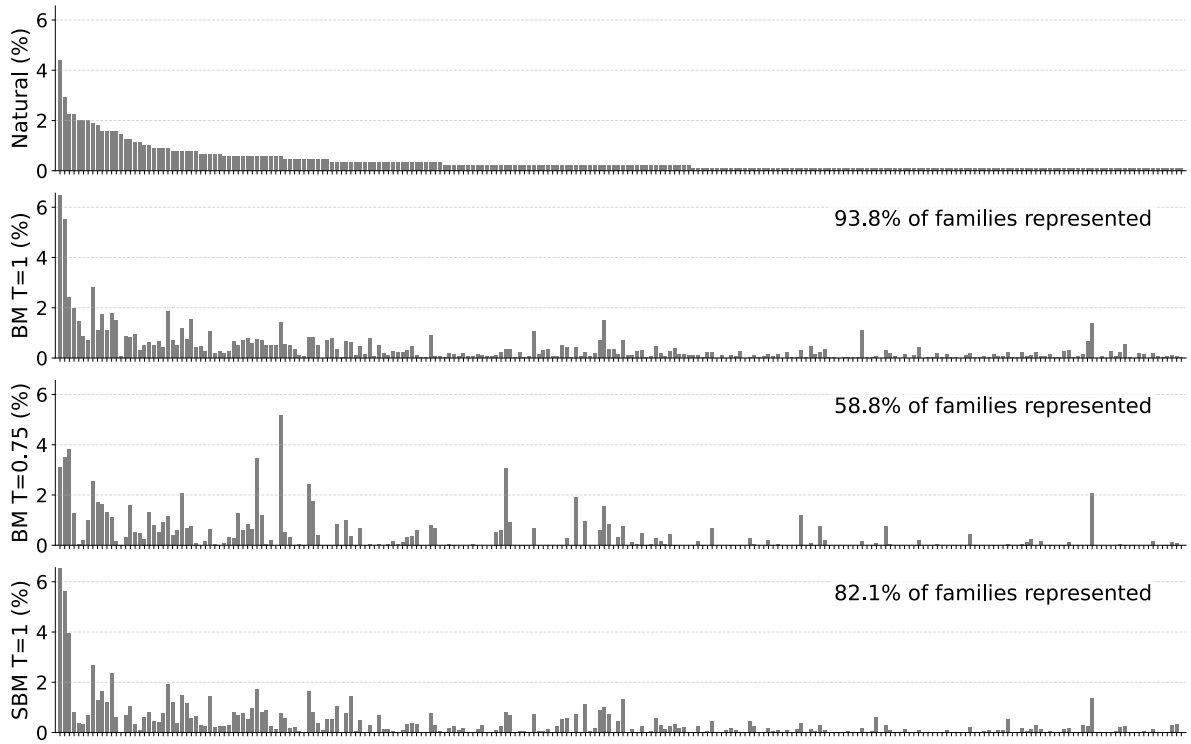

Figure C.6: CM family: Distribution of natural and artificial sequences across protein families.
The top panel reports the fraction of natural CM sequences that belong to each family, ranked from the most to the least frequent. This ordering is preserved along the x-axis of all panels: every bar represents one family found in the natural dataset. The lower panels display the same statistics for 10 000 artificial sequences generated with (i) a Boltzmann Machine (BM, $\lambda = 0.01$) at $T = 1$ and $T = 0.75$, and (ii) a Stochastic Boltzmann Machine (SBM, $N_{\text{chains}} = 50$) at $T = 1$. Each artificial sequence is assigned to the family of its closest natural counterpart. Reducing the temperature favours sampling a subset of families, thereby lowering overall diversity.

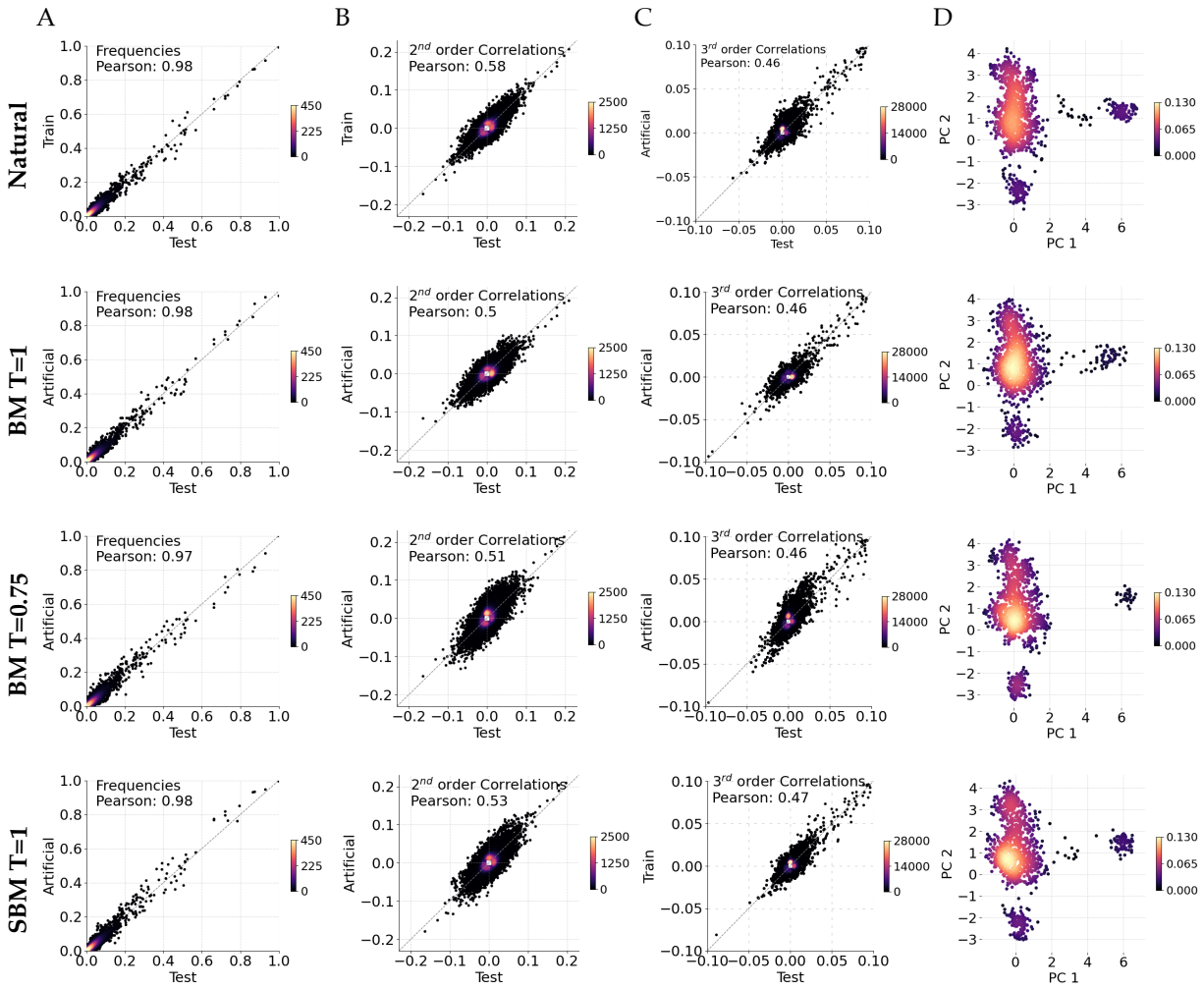

Figure C.7: CM family: Comparison of statistical properties between natural sequences and artificial sequences generated by SBM and BM models.

We compare statistics computed on artificial sequences generated by different models (rows) to those computed on the natural test set. The first row shows the comparison between the natural training and test data. Subsequent rows show results for Boltzmann Machines (BM) sampled at $T = 1$ and $T = 0.75$, and the SBM model at $T = 1$.

A. Comparison of single-site frequencies between test and artificial sequences. The inset shows the same comparison between training and test natural sequences. Pearson correlation coefficients are reported. Point density is indicated by color.

B. Same as panel A, but for 2^{nd}-order correlations.

C. Same as panel A, but for 3^{rd}-order correlations.

D. Projection of artificial sequences onto the first two principal components (PC1 and PC2) computed from the natural sequences. The inset shows the corresponding projection for the natural sequences themselves.

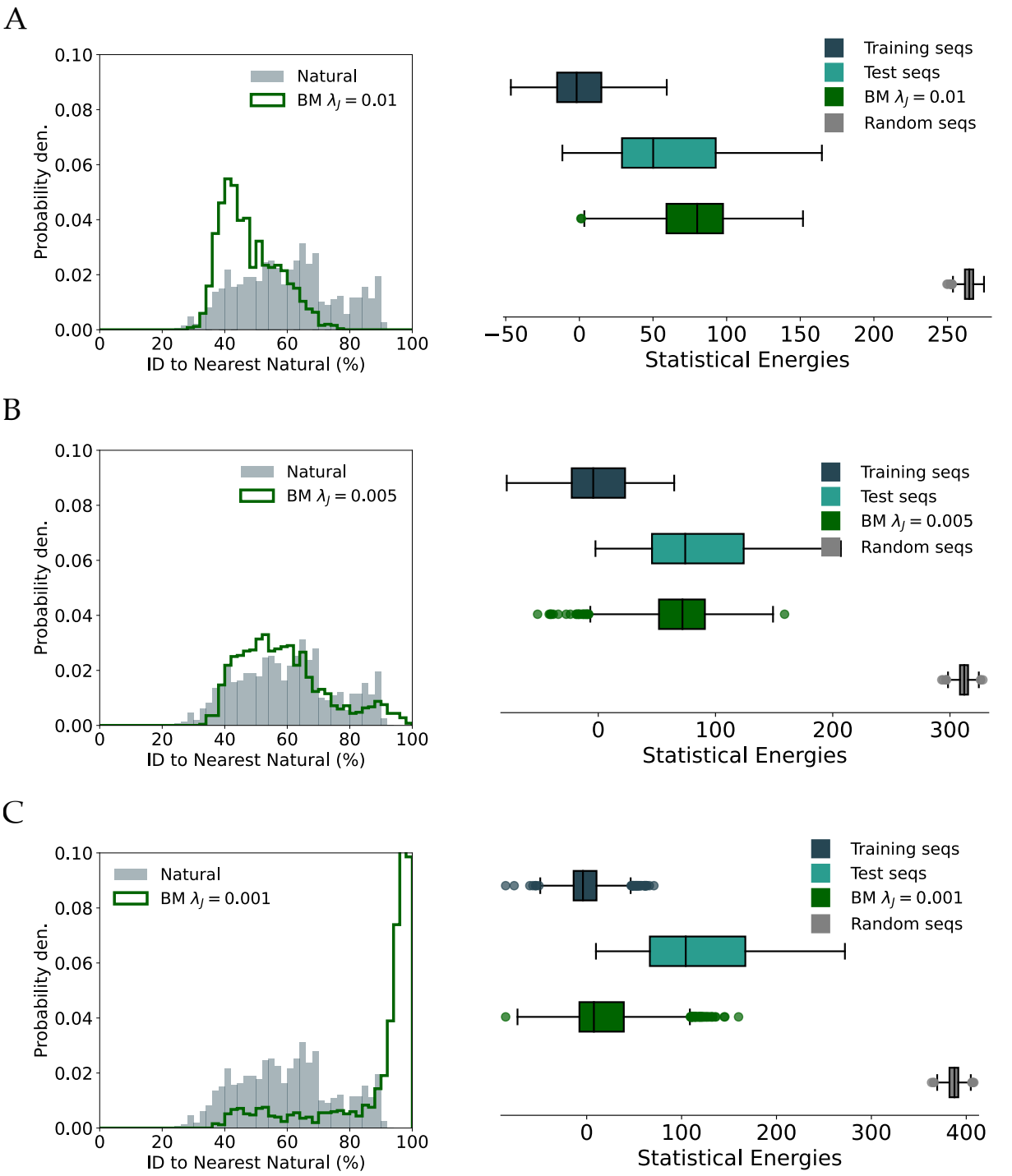

Figure C.8: CM family: Boltzmann-Machine (BM) models cannot simultaneously match natural energies and generate novel sequences.
Each panel reports statistics for sequences sampled at $T = 1$ from a BM trained on the CM family with an ℓ_2-regularization strength: **A**. $\lambda = 0.01$, **B**. $\lambda = 0.005$, **C**. $\lambda = 0.001$.
Left sub-panels: probability density of the percentage identity (ID) to the closest natural sequence (grey histogram: natural alignment; green curve: BM sequences).
Right sub-panels: box plots of statistical energies for training, test, BM-generated, and random sequences.
Lowering the regularization (smaller λ) drives the model to sample artificial sequences with statistical energies similar to training alignment, but at the cost of producing sequences that are almost all identical to natural ones (panel C). Conversely, stronger regularization (larger λ) yields more diverse sequences (panels A–B) but shifts their energy distribution away from that of the training data.

C.5 Results SBM $N_{\text{chains}} = 20$

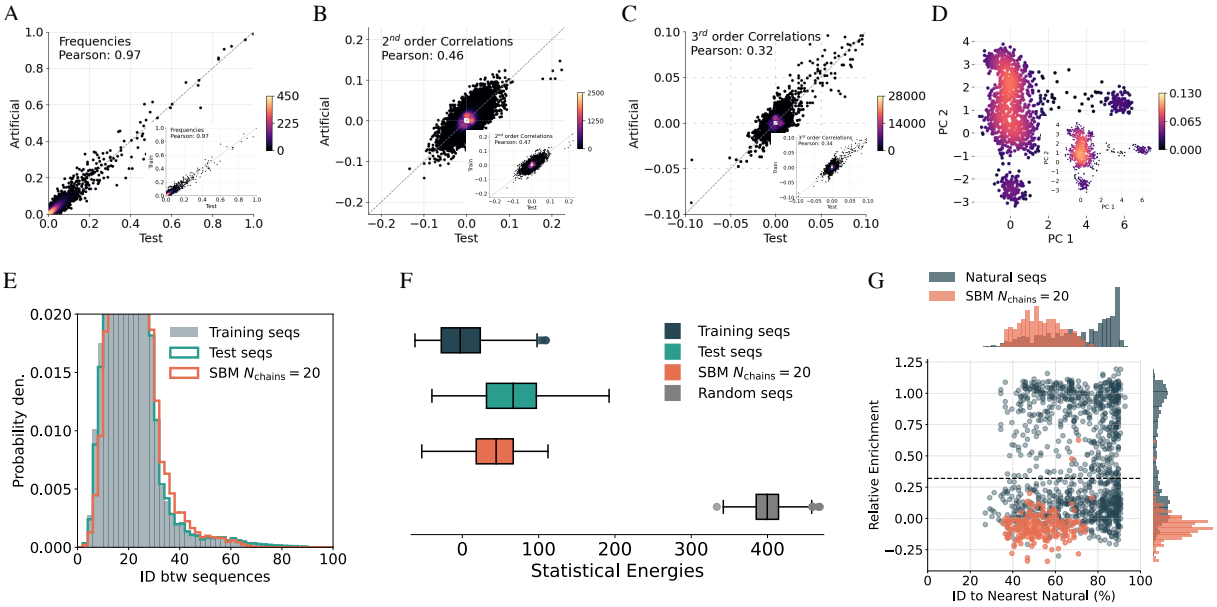

Figure C.9: SBM Statistics and generative power. (Pannel G is made from data provided by Emily H.)
A. Comparison of frequencies computed on the test and artificial sequences. The small pannel shows the same but comparing the training and test sequences. We also provide the Pearson coefficient. The colors shows the density of points.
B. Same as in pannel A, but we compare 2^{nd} correlations.
C. Same as in pannel A, but we compare 3^{rd} correlations.
D. Articifial sequences projected on the 2 principal components computed from the natural alignment. The small pannel show the same projection for the natural sequences for comparison.
E. Histograms of identities between sequences for Training sequences, Test sequences and artificial sequences generated with the SBM $N_{\text{chains}} = 30$.
F. Box plots showing statistical energy distributions for the traning sequences, test sequences, artificial sequences and random sequences.
G. Scatter plot showing showing the relative enrichment experimentally measured.

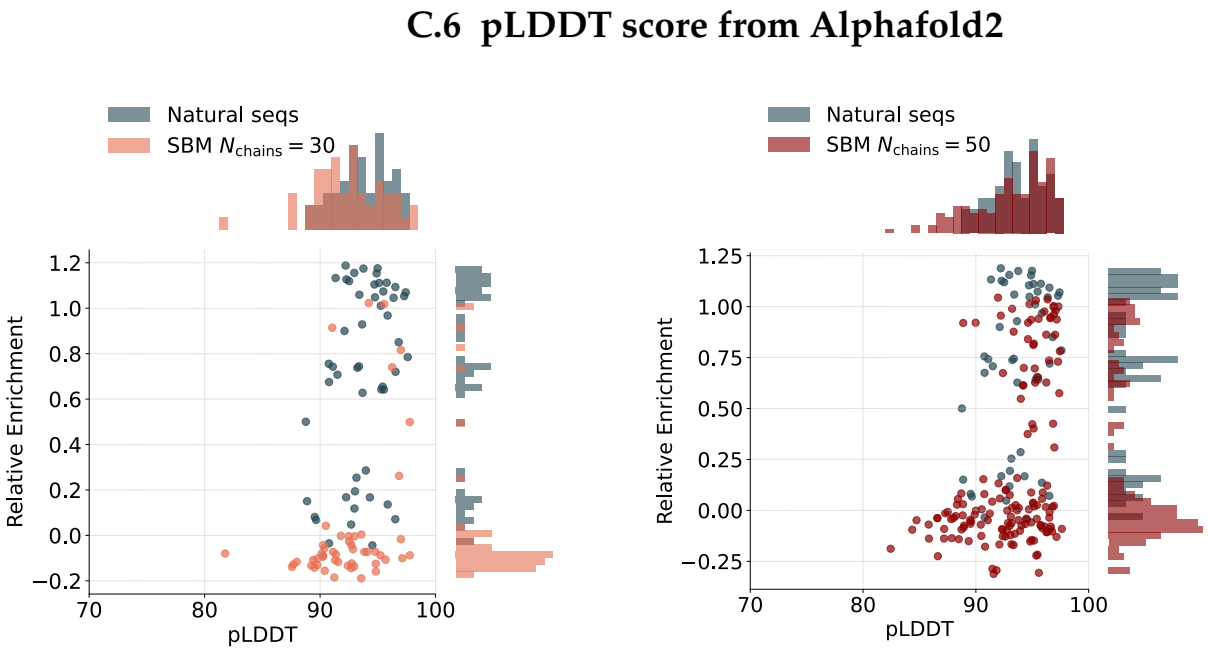

Figure C.10: Relative enrichment against pLDDT confidence score from AlphaFold2.
Comparison of the AlphaFold2 predicted confidence scores (pLDDT) (ColabFold version with default parameters) and relative enrichment for natural sequences (grey) and sequences generated using SBM with $N_{\text{chains}} = 30$ (left, orange) and $N_{\text{chains}} = 50$ (right, red). Marginal histograms display the distribution of pLDDT and enrichment values for each group.

Supplementary Serine Proteases D

D.1 Statistical Coupling Analysis on S1A family

We provide here a more comprehensive comparison of sector analyses performed using two different MSAs: one constructed by Shoichi Yip, containing 93,695 sequences, and another from Halabi *et al.* [19], which includes 1,470 sequences.

As shown in Figure D.1, the red and green sectors are robustly recovered across both alignments. In contrast, the blue sector identified by Halabi *et al.* is not recovered using the larger MSA. This reduced robustness of the blue sector has been previously noted by Olivier Rivoire in [32], where this sector was split into two distinct components.

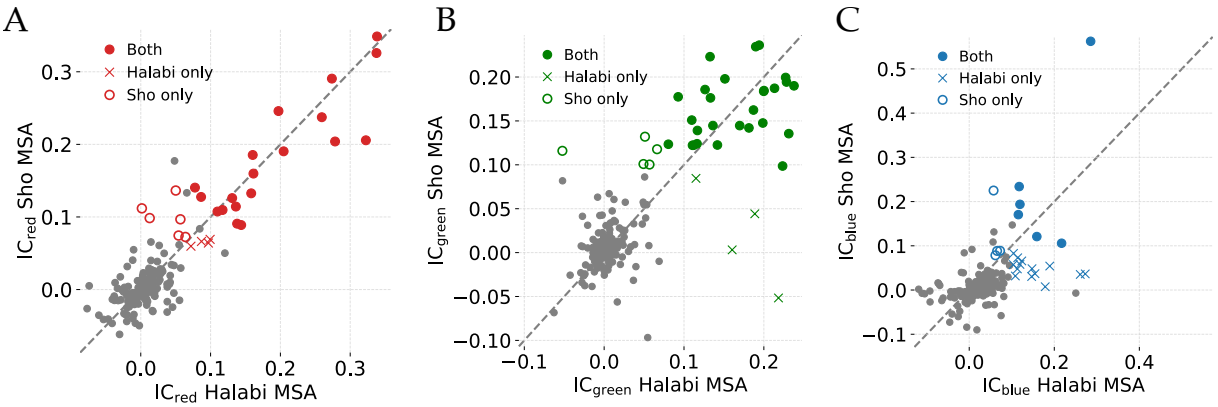

Figure D.1: SCA analysis using Shoichi's and Halabi's MSAs.
SCA analyses were performed using the COCOATREE toolbox [2].
A. Comparison of residue contributions to the red (specificity-related) sector obtained from two MSAs: the original alignment from Halabi *et al.* and the larger alignment built by Shoichi Yip. Red filled circles indicate residues identified in both MSAs; red crosses and open circles represent residues identified only in Halabi's or Shoichi's MSA, respectively.
B. Same as panel A, for the green sector.
C. Same as panel A, for the blue sector.

D.2 Experiments

We provide here additional informations about the experiments conducted by Amaury Paveyranne, Timothé Lucas et Shoichi Yip. For a more detailed description, please refer to Amaury Paveyranne's thesis [152].

The main steps of the experimental protocol are the following:

1. Each trypsin mutant are inserted into plasmids carrying a fluorescent protein mCherry that allow the quantification of expression levels.
2. The plasmids are transformed into *B. subtilis*.
3. Using IPTG[1], we induce the secretion of mutant by *B. subtilis* into the medium.
4. Samples of the culture medium are collected and mixed with a rhodamine-based substrate solution [155] in separate wells for each mutant (see Figure D.2).

1: IPTG is a molecule that makes the bacteria express the genes we have inserted.

5. Fluorescence measurements are performed in each well over time by tecan spark plate reader.

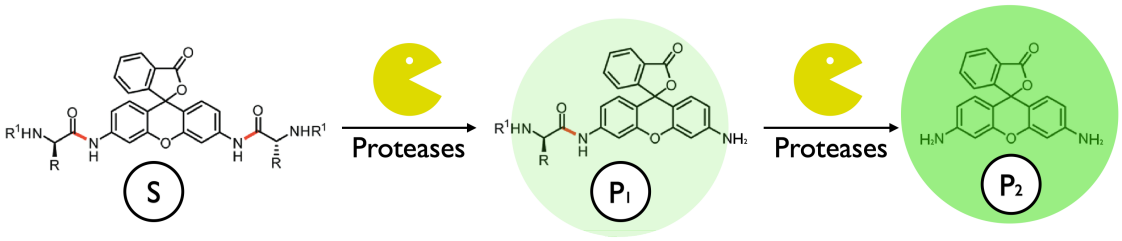

Figure D.2: Illustration of protease action on a rhodamine substrate. (Figure made by Amaury Paveyranne and adapted from Fenneteau *et al.* [171])
The rhodamine bisamide substrate (**S**, left) is essentially non-fluorescent. Cleavage of a single amide bond by a protease releases the monoamide intermediate ($\mathbf{P_1}$, centre), whose fluorescence is already markedly enhanced. A second cleavage event yields the fully deprotected rhodamine derivative ($\mathbf{P_2}$, right), producing a further increase in fluorescence.

D.3 Supplementary figures

D.3.1 Fluorescence measurements

In Figure D.3 we show the fluorescence measurements of all the mutants cited in Section 6.7 against a trypsin substrate. These results suggest that all the mutants tested have lost their trypsin activity. We also show the fluorescence measurements of all the mutants cited in Section 6.8 against a chymotrypsin substrate. These results suggest that all the mutants tested have lost their chymotrypsin activity.

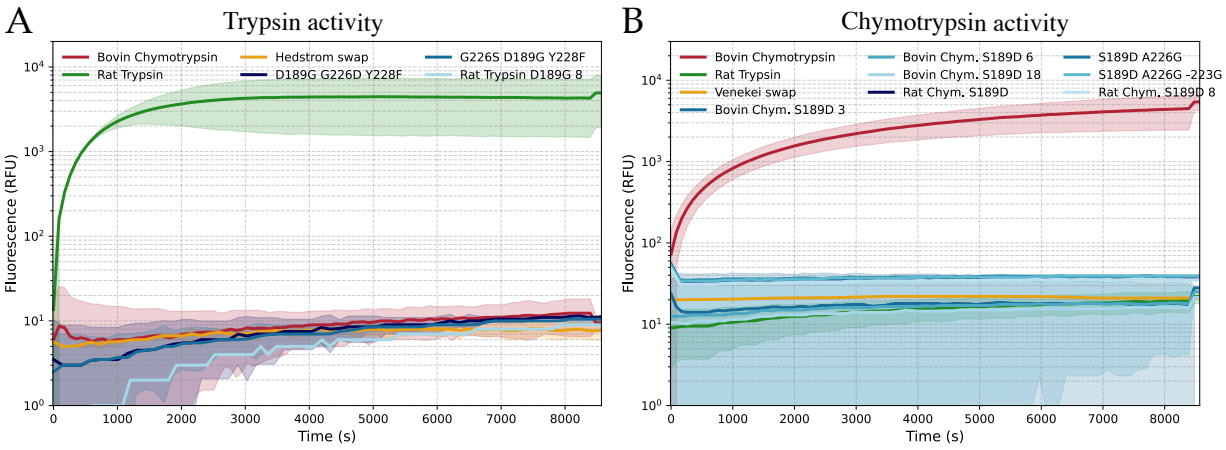

Figure D.3: Initial activity of COMBINE-generated mutants in specificity conversion assays.
D. Fluorescence curves over time for natural sequences and mutants tested on a Trypsin substrate. Rat Trypsin (green) serves as positive control, while Bovine Chymotrypsin (red) is the negative control. The orange curve corresponds to Hedstrom's swap [143], and the blue curves represent mutants proposed with the COMBINE method starting from rat trypsin.
E. Bovine Chymotrypsin (red) serves as positive control, while rat trypsin (green) is the negative control. The orange curve corresponds to Venekei's swap [145], and the blue curves represent successive mutants proposed by COMBINE starting from Bovine chymotrypsin or rat chymotrypsin B.

D.3.2 Impact of regulararization on COMBINE predictions

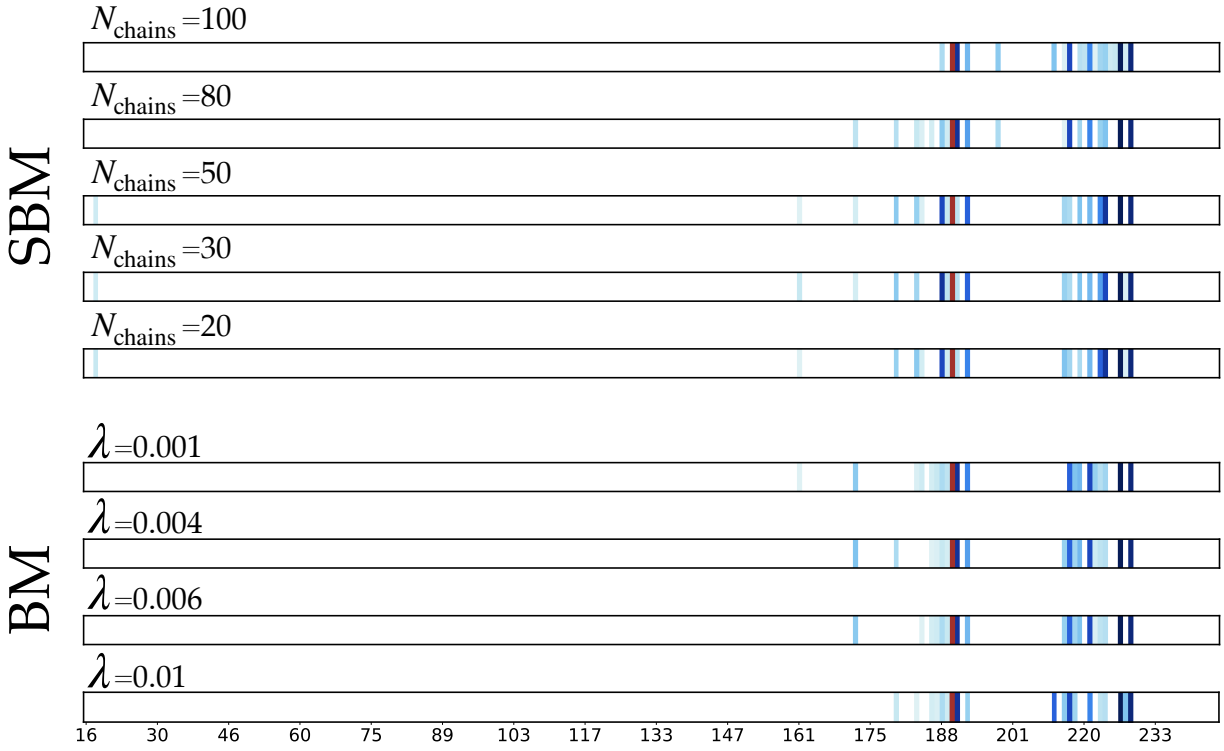

Figure D.4: Impact of regularization and inference method on COMBINE predictions.
Compensatory mutations predicted by the COMBINE method starting from the rat trypsin D189S mutant. Each horizontal bar represents a trypsin sequence, and vertical lines indicate the positions of compensatory mutations. The blue color gradient reflects the order in which mutations are proposed (from dark to light blue). The top panel shows results using models inferred with the SBM procedure for different levels of regularization, controlled by the number of MCMC chains N_{chains}. The bottom panel shows results obtained with standard Boltzmann Machines (BM) using varying levels of L_2 regularization λ.

Bibliography

[1] Paul Guenon et al. 'Allostery without Large Motions: Molecular Dissection of a Minimal-Shift MWC Allosteric Regulation'. in preparation. 2025 (cited on page 7).

[2] Margaux Julien et al. 'COCOA-Tree: COllaborative COevolution Analysis Toolbox'. in preparation. 2025 (cited on pages 8, 77, 123).

[3] Eugene V Koonin. *The logic of chance: the nature and origin of biological evolution*. FT press, 2011 (cited on page 11).

[4] Benjah-bmm27. *Structure of an Amino Acid*. Accessed: 2025-05-31. 2007. URL: https://fr.wikipedia.org/wiki/Acide_amin%C3%A9#/media/Fichier:Alpha-amino-acid-general-2D-stereo.png (cited on page 11).

[5] Bruce Alberts et al. *Molecular biology of the cell: seventh international student edition with registration card*. WW Norton & Company, 2022 (cited on pages 11, 13).

[6] Christian B Anfinsen et al. 'The kinetics of formation of native ribonuclease during oxidation of the reduced polypeptide chain'. In: *Proceedings of the National Academy of Sciences* 47.9 (1961), pp. 1309–1314 (cited on pages 12, 69).

[7] Christopher M Dobson. 'Protein folding and misfolding'. In: *Nature* 426.6968 (2003), pp. 884–890 (cited on page 12).

[8] Elena Papaleo et al. 'The role of protein loops and linkers in conformational dynamics and allostery'. In: *Chemical reviews* 116.11 (2016), pp. 6391–6423 (cited on page 12).

[9] 'UniProt: the Universal protein knowledgebase in 2025'. In: *Nucleic Acids Research* 53.D1 (2025), pp. D609–D617 (cited on pages 12–14).

[10] Warren L DeLano et al. 'Pymol: An open-source molecular graphics tool'. In: *CCP4 Newsl. Protein Crystallogr* 40.1 (2002), pp. 82–92 (cited on page 12).

[11] Peter W Rose et al. 'The RCSB protein data bank: integrative view of protein, gene and 3D structural information'. In: *Nucleic acids research* (2016), gkw1000 (cited on page 12).

[12] John C Kendrew et al. 'A three-dimensional model of the myoglobin molecule obtained by x-ray analysis'. In: *Nature* 181.4610 (1958), pp. 662–666 (cited on pages 12, 69).

[13] Kurt Wüthrich. 'The way to NMR structures of proteins'. In: *Nature structural biology* 8.11 (2001), pp. 923–925 (cited on pages 12, 69).

[14] John Jumper et al. 'Highly accurate protein structure prediction with AlphaFold'. In: *Nature* 596.7873 (2021), pp. 583–589 (cited on pages 13, 31, 69).

[15] Joseph Sambrook, Edward F Fritsch, and Tom Maniatis. *Molecular cloning: a laboratory manual*. Ed. 2. 1989 (cited on page 13).

[16] Sriram Kosuri and George M Church. 'Large-scale de novo DNA synthesis: technologies and applications'. In: *Nature methods* 11.5 (2014), pp. 499–507 (cited on page 13).

[17] Jay Shendure and Hanlee Ji. 'Next-generation DNA sequencing'. In: *Nature biotechnology* 26.10 (2008), pp. 1135–1145 (cited on pages 13, 14).

[18] Miten Jain et al. 'The Oxford Nanopore MinION: delivery of nanopore sequencing to the genomics community'. In: *Genome biology* 17 (2016), pp. 1–11 (cited on page 14).

[19] Najeeb Halabi et al. 'Protein sectors: evolutionary units of three-dimensional structure'. In: *Cell* 138.4 (2009), pp. 774–786 (cited on pages 15, 17, 18, 32, 40, 64, 77, 90, 102, 123).

[20] Christine A Orengo and Janet M Thornton. 'Protein families and their evolution—a structural perspective'. In: *Annu. Rev. Biochem.* 74.1 (2005), pp. 867–900 (cited on page 14).

[21] Enrico Di Cera. 'Serine proteases'. In: *IUBMB life* 61.5 (2009), pp. 510–515 (cited on pages 14, 75).

[22] Cyrus Chothia and Arthur M Lesk. 'The relation between the divergence of sequence and structure in proteins.' In: *The EMBO journal* 5.4 (1986), pp. 823–826 (cited on page 15).

[23] Matthias Blum et al. 'InterPro: the protein sequence classification resource in 2025<? mode longmeta?>' In: *Nucleic Acids Research* 53.D1 (2025), pp. D444–D456 (cited on pages 15, 69).

[24] Robert C Edgar and Serafim Batzoglou. 'Multiple sequence alignment'. In: *Current opinion in structural biology* 16.3 (2006), pp. 368–373 (cited on pages 15, 16).

[25] Kazutaka Katoh and Daron M Standley. 'MAFFT multiple sequence alignment software version 7: improvements in performance and usability'. In: *Molecular biology and evolution* 30.4 (2013), pp. 772–780 (cited on page 16).

[26] Robert C Edgar. 'MUSCLE: multiple sequence alignment with high accuracy and high throughput'. In: *Nucleic acids research* 32.5 (2004), pp. 1792–1797 (cited on page 16).

[27] Sean R Eddy. 'Accelerated profile HMM searches'. In: *PLoS computational biology* 7.10 (2011), e1002195 (cited on pages 16, 31).

[28] Julie D Thompson et al. 'A comprehensive benchmark study of multiple sequence alignment methods: current challenges and future perspectives'. In: *PloS one* 6.3 (2011), e18093 (cited on page 16).

[29] Olivier Rivoire, Kimberly A Reynolds, and Rama Ranganathan. 'Evolution-based functional decomposition of proteins'. In: *PLoS computational biology* 12.6 (2016), e1004817 (cited on pages 16–18, 40, 77, 90, 102).

[30] Yaakov Kleeorin et al. 'Undersampling and the inference of coevolution in proteins'. In: *bioRxiv* (2021), pp. 2021–04 (cited on pages 17, 25, 28, 32, 37, 39–43, 54, 55, 80, 101).

[31] Simona Cocco, Remi Monasson, and Martin Weigt. 'From principal component to direct coupling analysis of coevolution in proteins: low-eigenvalue modes are needed for structure prediction'. In: *PLoS computational biology* 9.8 (2013), e1003176 (cited on pages 17, 27).

[32] Olivier Rivoire. 'Elements of coevolution in biological sequences'. In: *Physical review letters* 110.17 (2013), p. 178102 (cited on pages 17, 18, 123).

[33] David T Jones et al. 'PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments'. In: *Bioinformatics* 28.2 (2012), pp. 184–190 (cited on pages 17, 38).

[34] Magnus Ekeberg et al. 'Improved contact prediction in proteins: using pseudolikelihoods to infer Potts models'. In: *Physical Review E* 87.1 (2013), p. 012707 (cited on pages 17, 26, 27, 38, 39).

[35] Stefan Seemayer, Markus Gruber, and Johannes Söding. 'CCMpred–fast and precise prediction of protein residue–residue contacts from correlated mutations'. In: *Bioinformatics* 30.21 (2014), pp. 3128–3130 (cited on page 17).

[36] Kimberly A Reynolds, Richard N McLaughlin, and Rama Ranganathan. 'Hot spots for allosteric regulation on protein surfaces'. In: *Cell* 147.7 (2011), pp. 1564–1575 (cited on pages 17, 18, 32, 40, 64, 90, 93, 94, 96).

[37] Debora S Marks, Thomas A Hopf, and Chris Sander. 'Protein structure prediction from sequence variation'. In: *Nature biotechnology* 30.11 (2012), pp. 1072–1080 (cited on page 17).

[38] Faruck Morcos et al. 'Direct-coupling analysis of residue coevolution captures native contacts across many protein families'. In: *Proceedings of the National Academy of Sciences* 108.49 (2011), E1293–E1301 (cited on pages 17, 27, 38, 39).

[39] Steve W Lockless and Rama Ranganathan. 'Evolutionarily conserved pathways of energetic connectivity in protein families'. In: *Science* 286.5438 (1999), pp. 295–299 (cited on pages 17, 93).

[40] Jean-Philippe Bouchaud and Marc Potters. *Theory of financial risk and derivative pricing: from statistical physics to risk management*. Cambridge university press, 2003 (cited on page 17).

[41] Ljubica Mihaljević and Siniša Urban. 'Decoding the functional evolution of an intramembrane protease superfamily by statistical coupling analysis'. In: *Structure* 28.12 (2020), pp. 1329–1336 (cited on pages 18, 32, 64).

[42] David H Ackley, Geoffrey E Hinton, and Terrence J Sejnowski. 'A learning algorithm for Boltzmann machines'. In: *Cognitive science* 9.1 (1985), pp. 147–169 (cited on page 21).

[43] Simona Cocco et al. 'Inverse statistical physics of protein sequences: a key issues review'. In: *Reports on Progress in Physics* 81.3 (2018), p. 032601 (cited on pages 22, 31, 39, 61, 63, 107).

[44] William Bialek et al. 'Statistical mechanics for natural flocks of birds'. In: *Proceedings of the National Academy of Sciences* 109.13 (2012), pp. 4786–4791 (cited on page 22).

[45] Thomas Bury. 'Market structure explained by pairwise interactions'. In: *Physica A: Statistical Mechanics and its Applications* 392.6 (2013), pp. 1375–1385 (cited on page 22).

[46] Edwin T Jaynes. 'Information theory and statistical mechanics'. In: *Physical review* 106.4 (1957), p. 620 (cited on page 22).

[47] Edwin T Jaynes. 'Information theory and statistical mechanics. II'. In: *Physical review* 108.2 (1957), p. 171 (cited on page 22).

[48] Juyong Park and Mark EJ Newman. 'Statistical mechanics of networks'. In: *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics* 70.6 (2004), p. 066117 (cited on page 22).

[49] Alan E Gelfand and Adrian FM Smith. 'Sampling-based approaches to calculating marginal densities'. In: *Journal of the American statistical association* 85.410 (1990), pp. 398–409 (cited on page 24).

[50] Nicholas Metropolis et al. 'Equation of state calculations by fast computing machines'. In: *The journal of chemical physics* 21.6 (1953), pp. 1087–1092 (cited on pages 24, 107).

[51] W Keith Hastings. 'Monte Carlo sampling methods using Markov chains and their applications'. In: (1970) (cited on pages 24, 107).

[52] Aurélien Decelle, Cyril Furtlehner, and Beatriz Seoane. 'Equilibrium and non-equilibrium regimes in the learning of restricted Boltzmann machines'. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 5345–5359 (cited on pages 24, 25).

[53] Miguel A Carreira-Perpinan and Geoffrey Hinton. 'On contrastive divergence learning'. In: *International workshop on artificial intelligence and statistics*. PMLR. 2005, pp. 33–40 (cited on page 25).

[54] Tijmen Tieleman. 'Training restricted Boltzmann machines using approximations to the likelihood gradient'. In: *Proceedings of the 25th international conference on Machine learning*. 2008, pp. 1064–1071 (cited on page 25).

[55] Edwin Rodriguez Horta and Martin Weigt. 'On the effect of phylogenetic correlations in coevolution-based contact prediction in proteins'. In: *PLoS computational biology* 17.5 (2021), e1008957 (cited on page 25).

[56] Martin Weigt et al. 'Identification of direct residue contacts in protein–protein interaction by message passing'. In: *Proceedings of the National Academy of Sciences* 106.1 (2009), pp. 67–72 (cited on pages 26, 27).

[57] William P Russ et al. 'Natural-like function in artificial WW domains'. In: *Nature* 437.7058 (2005), pp. 579–583 (cited on pages 26, 28, 30).

[58] William P Russ et al. 'An evolution-based model for designing chorismate mutase enzymes'. In: *Science* 369.6502 (2020), pp. 440–445 (cited on pages 26–28, 30, 31, 39, 43–45, 56, 58, 64, 101, 109, 110).

[59] Pierre Barrat-Charlaix et al. 'Sparse generative modeling via parameter reduction of Boltzmann machines: application to protein-sequence families'. In: *Physical Review E* 104.2 (2021), p. 024407 (cited on pages 26, 32, 38, 39).

[60] Francesco Calvanese et al. 'Towards parsimonious generative modeling of RNA families'. In: *Nucleic Acids Research* 52.10 (2024), pp. 5465–5477 (cited on pages 26, 38, 39).

[61] Alan S Lapedes et al. 'Correlated mutations in models of protein sequences: phylogenetic and structural effects'. In: *Lecture Notes-Monograph Series* (1999), pp. 236–256 (cited on page 27).

[62] Allan Haldane and Ronald M Levy. 'Mi3-GPU: MCMC-based inverse ising inference on GPUs for protein covariation analysis'. In: *Computer Physics Communications* 260 (2021), p. 107312 (cited on pages 27, 49).

[63] Timm Plefka. 'Convergence condition of the TAP equation for the infinite-ranged Ising spin glass model'. In: *Journal of Physics A: Mathematical and general* 15.6 (1982), p. 1971 (cited on page 27).

[64] Carlo Baldassi et al. 'Fast and accurate multivariate Gaussian modeling of protein families: predicting residue contacts and protein-interaction partners'. In: *PloS one* 9.3 (2014), e92721 (cited on page 27).

[65] Sivaraman Balakrishnan et al. 'Learning generative models for protein fold families'. In: *Proteins: Structure, Function, and Bioinformatics* 79.4 (2011), pp. 1061–1078 (cited on page 27).

[66] Jeanne Trinquier et al. 'Efficient generative modeling of protein sequences using simple autoregressive models'. In: *Nature communications* 12.1 (2021), p. 5800 (cited on pages 27, 28, 31, 61, 63).

[67] Thomas A Hopf et al. 'Sequence co-evolution gives 3D contacts and structures of protein complexes'. In: *elife* 3 (2014), e03430 (cited on page 27).

[68] Sergey Ovchinnikov, Hetunandan Kamisetty, and David Baker. 'Robust and accurate prediction of residue–residue interactions across protein interfaces using evolutionary information'. In: *elife* 3 (2014), e02030 (cited on page 27).

[69] Ulrike Göbel et al. 'Correlated mutations and residue contacts in proteins'. In: *Proteins: Structure, Function, and Bioinformatics* 18.4 (1994), pp. 309–317 (cited on page 28).

[70] Neil D Clarke. 'Covariation of residues in the homeodomain sequence family'. In: *Protein Science* 4.11 (1995), pp. 2269–2278 (cited on page 28).

[71] Lukas Burger and Erik Van Nimwegen. 'Disentangling direct from indirect co-evolution of residues in protein alignments'. In: *PLoS computational biology* 6.1 (2010), e1000633 (cited on page 28).

[72] Anne-Florence Bitbol et al. 'Inferring interaction partners from protein sequences'. In: *Proceedings of the National Academy of Sciences* 113.43 (2016), pp. 12180–12185 (cited on page 28).

[73] Thomas Gueudré et al. 'Simultaneous identification of specifically interacting paralogs and interprotein contacts by direct coupling analysis'. In: *Proceedings of the National Academy of Sciences* 113.43 (2016), pp. 12186–12191 (cited on page 28).

[74] Hervé Jacquier et al. 'Capturing the mutational landscape of the beta-lactamase TEM-1'. In: *Proceedings of the National Academy of Sciences* 110.32 (2013), pp. 13067–13072 (cited on pages 28, 72).

[75] Matteo Figliuzzi et al. 'Coevolutionary landscape inference and the context-dependence of mutations in beta-lactamase TEM-1'. In: *Molecular biology and evolution* 33.1 (2016), pp. 268–280 (cited on pages 28, 38, 39, 73, 80).

[76] Ryan R Cheng et al. 'Connecting the sequence-space of bacterial signaling proteins to phenotypes using coevolutionary landscapes'. In: *Molecular biology and evolution* 33.12 (2016), pp. 3054–3064 (cited on pages 28, 80).

[77] Thomas A Hopf et al. 'Mutation effects predicted from sequence co-variation'. In: *Nature biotechnology* 35.2 (2017), pp. 128–135 (cited on pages 28, 39, 73, 80).

[78] Michael Socolich et al. 'Evolutionary information for specifying a protein fold'. In: *Nature* 437.7058 (2005), pp. 512–518 (cited on page 28).

[79] Mehrsa Mardikoraem et al. 'Generative models for protein sequence modeling: recent advances and future directions'. In: *Briefings in Bioinformatics* 24.6 (2023), bbad358 (cited on page 28).

[80] Alex Hawkins-Hooker et al. 'Generating functional protein variants with variational autoencoders'. In: *PLoS computational biology* 17.2 (2021), e1008736 (cited on pages 28–31, 61, 63).

[81] Niksa Praljak et al. 'ProtWave-VAE: Integrating autoregressive sampling with latent-based inference for data-driven protein design'. In: *ACS synthetic biology* 12.12 (2023), pp. 3544–3561 (cited on pages 28, 30).

[82] Diederik P Kingma, Max Welling, et al. *Auto-encoding variational bayes*. 2013 (cited on page 29).

[83] Ian J Goodfellow et al. 'Generative adversarial nets'. In: *Advances in neural information processing systems* 27 (2014) (cited on page 29).

[84] Donatas Repecka et al. 'Expanding functional protein sequence spaces using generative adversarial networks'. In: *Nature Machine Intelligence* 3.4 (2021), pp. 324–333 (cited on page 29).

[85] Tianyang Lin et al. 'A survey of transformers'. In: *AI open* 3 (2022), pp. 111–132 (cited on page 29).

[86] Ashish Vaswani et al. 'Attention is all you need'. In: *Advances in neural information processing systems* 30 (2017) (cited on page 29).

[87] Zachary Wu et al. 'Signal peptides generated by attention-based neural networks'. In: *ACS Synthetic Biology* 9.8 (2020), pp. 2154–2161 (cited on page 29).

[88] Damiano Sgarbossa, Umberto Lupo, and Anne-Florence Bitbol. 'Generative power of a protein language model trained on multiple sequence alignments'. In: *Elife* 12 (2023), e79854 (cited on pages 29, 31, 32, 61, 63).

[89] Noelia Ferruz, Steffen Schmidt, and Birte Höcker. 'ProtGPT2 is a deep unsupervised language model for protein design'. In: *Nature communications* 13.1 (2022), p. 4348 (cited on pages 29, 31).

[90] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 'Denoising diffusion probabilistic models'. In: *Advances in neural information processing systems* 33 (2020), pp. 6840–6851 (cited on page 29).

[91] Joseph L Watson et al. 'De novo design of protein structure and function with RFdiffusion'. In: *Nature* 620.7976 (2023), pp. 1089–1100 (cited on pages 29, 30).

[92] A Alaa et al. *How faithful is your synthetic data? Sample-level metrics for evaluating and auditing generative models. 2021* (cited on pages 29, 63).

[93] Trevor Hastie, Robert Tibshirani, Jerome Friedman, et al. *The elements of statistical learning*. 2009 (cited on pages 30, 38).

[94] Jung-Eun Shin et al. 'Protein design and variant prediction using autoregressive generative models'. In: *Nature communications* 12.1 (2021), p. 2403 (cited on page 30).

[95] Rosalie Lipsh-Sokolik et al. 'Combinatorial assembly and design of enzymes'. In: *Science* 379.6628 (2023), pp. 195–201 (cited on page 30).

[96] Kai Shimagaki and Martin Weigt. 'Selection of sequence motifs and generative Hopfield-Potts models for protein families'. In: *Physical Review E* 100.3 (2019), p. 032128 (cited on pages 31, 61, 63).

[97] Allan Haldane et al. 'Coevolutionary landscape of kinase family proteins: sequence probabilities and functional motifs'. In: *Biophysical journal* 114.1 (2018), pp. 21–31 (cited on pages 31, 38, 61, 63).

[98] Bo Ni, David L Kaplan, and Markus J Buehler. 'Generative design of de novo proteins based on secondary-structure constraints using an attention-based diffusion model'. In: *Chem* 9.7 (2023), pp. 1828–1849 (cited on page 31).

[99] Andrew Leaver-Fay et al. 'ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules'. In: *Methods in enzymology*. Vol. 487. Elsevier, 2011, pp. 545–574 (cited on page 31).

[100] Enrico Ventura et al. 'Unlearning regularization for Boltzmann machines'. In: *Machine Learning: Science and Technology* 5.2 (2024), p. 025078 (cited on page 32).

[101] Allan Haldane and Ronald M Levy. 'Influence of multiple-sequence-alignment depth on Potts statistical models of protein covariation'. In: *Physical Review E* 99.3 (2019), p. 032405 (cited on page 32).

[102] Francisco McGee et al. 'The generative capacity of probabilistic protein sequence models'. In: *Nature communications* 12.1 (2021), p. 6302 (cited on pages 32, 63).

[103] Umberto Lupo, Damiano Sgarbossa, and Anne-Florence Bitbol. 'Protein language models trained on multiple sequence alignments learn phylogenetic relationships'. In: *Nature Communications* 13.1 (2022), p. 6298 (cited on pages 32, 63).

[104] Chen-Yi Gao, Hai-Jun Zhou, and Erik Aurell. 'Correlation-compressed direct-coupling analysis'. In: *Physical Review E* 98.3 (2018), p. 032407 (cited on pages 38, 39).

[105] Francesca Rizzato et al. 'Inference of compressed Potts graphical models'. In: *Physical Review E* 101.1 (2020), p. 012309 (cited on page 38).

[106] Lynne Reed Murphy, Anders Wallqvist, and Ronald M Levy. 'Simplified amino acid alphabets for protein fold recognition and implications for folding'. In: *Protein engineering* 13.3 (2000), pp. 149–152 (cited on page 38).

[107] Simona Cocco, Lorenzo Posani, and Rémi Monasson. 'Functional effects of mutations in proteins can be predicted and interpreted by guided selection of sequence covariation information'. In: *Proceedings of the National Academy of Sciences* 121.26 (2024), e2312335121 (cited on page 39).

[108] David H Calhoun et al. 'The emerging periplasm-localized subclass of AroQ chorismate mutases, exemplified by those from Salmonella typhimurium and Pseudomonas aeruginosa'. In: *Genome Biology* 2 (2001), pp. 1–16 (cited on page 44).

[109] Ulisse Ferrari. 'Learning maximum entropy models from finite-size data sets: A fast data-driven algorithm allows sampling from the posterior distribution'. In: *Physical Review E* 94.2 (2016), p. 023301 (cited on page 48).

[110] Giovanni Catania et al. 'A theoretical framework for overfitting in energy-based modeling'. In: *arXiv preprint arXiv:2501.19158* (2025) (cited on page 49).

[111] Andrew L Ferguson et al. 'Translating HIV sequences into quantitative fitness landscapes predicts viral vulnerabilities for rational immunogen design'. In: *Immunity* 38.3 (2013), pp. 606–617 (cited on page 49).

[112] Anna Paola Muntoni et al. 'adabmDCA: adaptive Boltzmann machine learning for biological sequences'. In: *BMC bioinformatics* 22.1 (2021), pp. 1–19 (cited on page 49).

[113] Jorge Nocedal and Stephen J Wright. *Numerical optimization*. Springer, 1999 (cited on pages 49–51, 113).

[114] Jorge Nocedal. 'Updating quasi-Newton matrices with limited storage'. In: *Mathematics of computation* 35.151 (1980), pp. 773–782 (cited on page 51).

[115] Ian Goodfellow et al. *Deep learning*. Vol. 1. 2. MIT press Cambridge, 2016 (cited on page 52).

[116] Léon Bottou. 'Large-scale machine learning with stochastic gradient descent'. In: *Proceedings of COMPSTAT'2010: 19th International Conference on Computational StatisticsParis France, August 22-27, 2010 Keynote, Invited and Contributed Papers*. Springer. 2010, pp. 177–186 (cited on page 53).

[117] Marina A Pak et al. 'Using AlphaFold to predict the impact of single mutations on protein stability and function'. In: *Plos one* 18.3 (2023), e0282689 (cited on page 63).

[118] Lucas Theis, Aäron van den Oord, and Matthias Bethge. 'A note on the evaluation of generative models'. In: *arXiv preprint arXiv:1511.01844* (2015) (cited on page 63).

[119] Kiera H Sumida et al. 'Improving protein expression, stability, and function with ProteinMPNN'. In: *Journal of the American Chemical Society* 146.3 (2024), pp. 2054–2061 (cited on page 64).

[120] Zachary Wu et al. 'Machine learning-assisted directed protein evolution with combinatorial libraries'. In: *Proceedings of the National Academy of Sciences* 116.18 (2019), pp. 8852–8858 (cited on page 64).

[121] Riccardo Rende et al. 'Mapping of attention mechanisms to a generalized potts model'. In: *Physical Review Research* 6.2 (2024), p. 023057 (cited on page 65).

[122] Christopher M Johnson. 'Differential scanning calorimetry as a tool for protein folding and stability'. In: *Archives of biochemistry and biophysics* 531.1-2 (2013), pp. 100–109 (cited on page 69).

[123] Kenneth A Johnson and Roger S Goody. 'The original Michaelis constant: translation of the 1913 Michaelis–Menten paper'. In: *Biochemistry* 50.39 (2011), pp. 8264–8269 (cited on page 70).

[124] Douglas M Fowler and Stanley Fields. 'Deep mutational scanning: a new style of protein science'. In: *Nature methods* 11.8 (2014), pp. 801–807 (cited on page 71).

[125] C Anders Olson, Nicholas C Wu, and Ren Sun. 'A comprehensive biophysical description of pairwise epistasis throughout an entire protein domain'. In: *Current biology* 24.22 (2014), pp. 2643–2651 (cited on pages 72, 84).

[126] Claudia Bank et al. 'A systematic survey of an intragenic epistatic landscape'. In: *Molecular biology and evolution* 32.1 (2015), pp. 229–238 (cited on page 72).

[127] William Bateson. 'Facts limiting the theory of heredity'. In: *Science* 26.672 (1907), pp. 649–660 (cited on page 72).

[128] Patrick C Phillips. 'Epistasis—the essential role of gene interactions in the structure and evolution of genetic systems'. In: *Nature Reviews Genetics* 9.11 (2008), pp. 855–867 (cited on page 72).

[129] Frank J Poelwijk, Vinod Krishna, and Rama Ranganathan. 'The context-dependence of mutations: a linkage of formalisms'. In: *PLoS computational biology* 12.6 (2016), e1004771 (cited on page 72).

[130] Tyler N Starr and Joseph W Thornton. 'Epistasis in protein evolution'. In: *Protein science* 25.7 (2016), pp. 1204–1218 (cited on page 72).

[131] Milo S Johnson, Gautam Reddy, and Michael M Desai. 'Epistasis and evolution: recent advances and an outlook for prediction'. In: *BMC biology* 21.1 (2023), p. 120 (cited on page 72).

[132] Frank J Poelwijk, Michael Socolich, and Rama Ranganathan. 'Learning the pattern of epistasis linking genotype and phenotype in a protein'. In: *Nature communications* 10.1 (2019), p. 4213 (cited on pages 72, 84, 96).

[133] Anna I Podgornaia and Michael T Laub. 'Pervasive degeneracy and epistasis in a protein-protein interface'. In: *Science* 347.6222 (2015), pp. 673–677 (cited on pages 72, 84).

[134] Wanzhi Huang and Timothy Palzkill. 'A natural polymorphism in β-lactamase is a global suppressor'. In: *Proceedings of the National Academy of Sciences* 94.16 (1997), pp. 8801–8806 (cited on page 72).

[135] Ngak-Leng Sim et al. 'SIFT web server: predicting effects of amino acid substitutions on proteins'. In: *Nucleic acids research* 40.W1 (2012), W452–W457 (cited on page 73).

[136] Elodie Laine, Yasaman Karami, and Alessandra Carbone. 'GEMME: a simple and fast global epistatic model predicting mutational effects'. In: *Molecular biology and evolution* 36.11 (2019), pp. 2604–2619 (cited on page 73).

[137] Adam J Riesselman, John B Ingraham, and Debora S Marks. 'Deep generative models of genetic variation capture the effects of mutations'. In: *Nature methods* 15.10 (2018), pp. 816–822 (cited on page 73).

[138] Pauline Hermans et al. 'Exploring evolution to uncover insights into protein mutational stability'. In: *Molecular Biology and Evolution* 42.1 (2025), msae267 (cited on page 73).

[139] Pascal Notin et al. 'Proteingym: Large-scale benchmarks for protein fitness prediction and design'. In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 64331–64379 (cited on page 73).

[140] Alan J Barrett, J Fred Woessner, and Neil D Rawlings. *Handbook of Proteolytic Enzymes, Volume 1*. Elsevier, 2012 (cited on page 75).

[141] Lizbeth Hedstrom. 'Serine protease mechanism and specificity'. In: *Chemical reviews* 102.12 (2002), pp. 4501–4524 (cited on page 75).

[142] Lizbeth Hedstrom, Laszlo Szilagyi, and William J Rutter. 'Converting trypsin to chymotrypsin: the role of surface loops'. In: *Science* 255.5049 (1992), pp. 1249–1253 (cited on pages 75, 76).

[143] Lizbeth Hedstrom, John J Perona, and William J Rutter. 'Converting trypsin to chymotrypsin: residue 172 is a substrate specificity determinant'. In: *Biochemistry* 33.29 (1994), pp. 8757–8763 (cited on pages 76, 77, 102, 124).

[144] Antonio Caputo et al. 'Conversion of the substrate specificity of mouse proteinase granzyme B'. In: *Nature structural biology* 1.6 (1994), pp. 364–367 (cited on page 76).

[145] István Venekei et al. 'Attempts to convert chymotrypsin to trypsin'. In: *FEBS letters* 379.2 (1996), pp. 143–147 (cited on pages 76, 88, 124).

[146] Su-Hwi Hung and Lizbeth Hedstrom. 'Converting trypsin to elastase: substitution of the S1 site and adjacent loops reconstitutes esterase specificity but not amidase activity.' In: *Protein engineering* 11.8 (1998), pp. 669–673 (cited on page 76).

[147] Antonio Caputo et al. 'Electrostatic reversal of serine proteinase substrate specificity'. In: *Proteins: Structure, Function, and Bioinformatics* 35.4 (1999), pp. 415–424 (cited on page 76).

[148] Balázs Jelinek et al. 'Ala226 to Gly and Ser189 to Asp mutations convert rat chymotrypsin B to a trypsin-like protease'. In: *Protein Engineering Design and Selection* 17.2 (2004), pp. 127–131 (cited on page 76).

[149] László Gráf et al. 'Electrostatic complementarity within the substrate-binding pocket of trypsin.' In: *Proceedings of the National Academy of Sciences* 85.14 (1988), pp. 4961–4965 (cited on pages 76, 79).

[150] Michael J Page et al. 'Conversion of trypsin into a Na+-activated enzyme'. In: *Biochemistry* 45.9 (2006), pp. 2987–2993 (cited on pages 76, 91).

[151] Navin Varadarajan et al. 'Highly active and selective endopeptidases with programmed substrate specificities'. In: *Nature chemical biology* 4.5 (2008), pp. 290–294 (cited on page 76).

[152] Paveyranne Amaury. 'Ingénierie d'enzymes via la génération à haut débit de données expériementales in vitro et leur modélisation statistiques'. in preparation. 2025 (cited on pages 80, 81, 84, 85, 123).

[153] Thuy N Nguyen et al. 'The genetic landscape of a metabolic interaction'. In: *Nature communications* 15.1 (2024), p. 3351 (cited on page 80).

[154] Valentin Senlis. 'Criblage à haut débit de l'effet des mutations sur l'activité catalytique et la spécificité de substrat de la trypsine'. PhD thesis. Université Paris sciences et lettres, 2021 (cited on page 84).

[155] Steven P Leytus, William L Patterson, and Walter F Mangel. 'New class of sensitive and selective fluorogenic substrates for serine proteinases. Amino acid and dipeptide derivatives of rhodamine'. In: *Biochemical Journal* 215.2 (1983), pp. 253–260 (cited on pages 85, 123).

[156] Nina M Goodey and Stephen J Benkovic. 'Allosteric regulation and catalysis emerge via a common route'. In: *Nature chemical biology* 4.8 (2008), pp. 474–482 (cited on page 93).

[157] ER Stadtman. 'Allosteric regulation of enzyme activity'. In: *Adv Enzymol Relat Areas Mol Biol* 28 (1966), pp. 41–154 (cited on page 93).

[158] Sherwin S Lehrer and Michael A Geeves. 'The muscle thin filament as a classical cooperative/allosteric regulatory system'. In: *Journal of molecular biology* 277.5 (1998), pp. 1081–1089 (cited on page 93).

[159] Richard B Vallee, Richard J McKenney, and Kassandra M Ori-McKenney. 'Multiple modes of cytoplasmic dynein regulation'. In: *Nature cell biology* 14.3 (2012), pp. 224–230 (cited on page 93).

[160] J. Monod, J. Wyman, and J. P. Changeux. 'On the nature of allosteric transitions: a plausible model'. In: *J. Mol. Biol.* 12.1 (1965), pp. 88–118 (cited on page 93).

[161] Daniel E Koshland Jr, George Némethy, and David Filmer. 'Comparison of experimental binding data and theoretical models in proteins containing subunits'. In: *Biochemistry* 5.1 (1966), pp. 365–385 (cited on page 93).

[162] Michael W Clarkson et al. 'Dynamic coupling and allosteric behavior in a nonallosteric protein'. In: *Biochemistry* 45.25 (2006), pp. 7693–7699 (cited on page 93).

[163] Maria Valeria Raimondi et al. 'DHFR inhibitors: reading the past for discovering novel anticancer agents'. In: *Molecules* 24.6 (2019), p. 1140 (cited on page 94).

[164] Craig E Cameron and Stephen J Benkovic. 'Evidence for a functional role of the dynamics of glycine-121 of Escherichia coli dihydrofolate reductase obtained from kinetic analysis of a site-directed mutant'. In: *Biochemistry* 36.50 (1997), pp. 15792–15800 (cited on pages 94, 95, 98).

[165] PT Ravi Rajagopalan, Stefan Lutz, and Stephen J Benkovic. 'Coupling interactions of distal residues enhance dihydrofolate reductase catalysis: mutational effects on hydride transfer rates'. In: *Biochemistry* 41.42 (2002), pp. 12618–12628 (cited on page 94).

[166] James W McCormick et al. 'Structurally distributed surface sites tune allosteric regulation'. In: *elife* 10 (2021), e68346 (cited on page 94).

[167] Shintaroh Kubo, Wenfei Li, and Shoji Takada. 'Allosteric conformational change cascade in cytoplasmic dynein revealed by structure-based molecular simulations'. In: *PLoS computational biology* 13.9 (2017), e1005748 (cited on page 94).

[168] Ron O Dror et al. 'Pathway and mechanism of drug binding to G-protein-coupled receptors'. In: *Proceedings of the National Academy of Sciences* 108.32 (2011), pp. 13118–13123 (cited on page 94).

[169] Yuji Sugita and Yuko Okamoto. 'Replica-exchange molecular dynamics method for protein folding'. In: *Chemical physics letters* 314.1-2 (1999), pp. 141–151 (cited on page 95).

[170] Thomas L Kalmer et al. 'Statistical Coupling Analysis Predicts Correlated Motions in Dihydrofolate Reductase'. In: *The Journal of Physical Chemistry B* 128.42 (2024), pp. 10373–10384 (cited on pages 96, 97).

[171] Johan Fenneteau et al. 'Synthesis of new hydrophilic rhodamine based enzymatic substrates compatible with droplet-based microfluidic assays'. In: *Chemical Communications* 53.39 (2017), pp. 5437–5440 (cited on page 124).