Statistical Physics - Machine Learning Interplay : from Addressing Class Imbalance with Replica Theory to Predicting Dynamical Heterogeneities with SE(3)-equivariant Graph Neural Networks

Francesco Pezzicoli

▶ **To cite this version:**

Francesco Pezzicoli. Statistical Physics - Machine Learning Interplay : from Addressing Class Imbalance with Replica Theory to Predicting Dynamical Heterogeneities with SE(3)-equivariant Graph Neural Networks. Machine Learning [cs.LG]. Université Paris-Saclay, 2024. English. NNT : 2024UPASG115 . tel-04910839

HAL Id: tel-04910839
https://theses.hal.science/tel-04910839v1

Submitted on 24 Jan 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

université
PARIS-SACLAY

Statistical Physics - Machine Learning Interplay: from Addressing Class Imbalance with Replica Theory to Predicting Dynamical Heterogeneities with SE(3)-equivariant Graph Neural Networks

Interactions entre Physique Statistique et Apprentissage Automatique: de l'étude du Déséquilibre de Classe avec la Méthode des Répliques à la Prédiction des Hétérogénéités Dynamiques avec des Réseaux de Neurones sur Graphes SE(3)-équivariants.

Thèse de doctorat de l'Université Paris-Saclay

École doctorale n° 580, sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat: Informatique mathématique
Graduate School: Informatique et sciences du numérique. Référent: Faculté des sciences d'Orsay

Thèse préparée dans l'unité de recherche **LISN** (CNRS, Université Paris-Saclay, INRIA, centraleSupélec), sous la direction de **Michèle SEBAG**, Directrice de Recherche (CNRS), le co-encadrement de **François LANDES**, Maitre des conférences (Université Paris-Saclay) et le co-encadrement de **Guillaume CHARPIAT**, Chargé de Recherche (INRIA)

Thèse présentée et soutenue à Paris-Saclay, le 19 Décembre 2024, par

Francesco Saverio PEZZICOLI

Composition du jury

Membres du jury avec voix délibérative

Martin Weigt Professeur des universités, LCQB, Sorbonne Université, CNRS	President
Sergei Grudinin Chargé de Recherche, HDR, LJK, Université Grenoble Alpes, CNRS	Rapporteur & Examinateur
Alejandro Rodriguez Garcia Professeur assistant, HDR, Univeristà di Trieste	Rapporteur & Examinateur
Nataliya Sokolovska Professeur des universités, LCQB, Sorbonne Université, CNRS	Examinatrice
Damien Vandembroucq Directeur de Recherche, PMMH, ESPCI, CNRS	Examinateur

THESE DE DOCTORAT
NNT: 2024UPASG115

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre: Interactions entre Physique Statistique et Apprentissage Automatique: de l'étude du Déséquilibre de Classe avec la Méthode des Répliques à la Prédiction des Hétérogénéités Dynamiques avec des Réseaux de Neurones sur Graphes SE(3)-équivariants.

Mots clés: Apprentissage Automatique, Données Déséquilibrées, Méthode des Répliques, Verres Structuraux, Réseaux de Neurones à Graphes

Résumé: Cette thèse explore la relation entre l'Apprentissage Automatique (AA) et la Physique Statistique (PS), en abordant deux défis importants à l'interface entre ces deux domaines. Tout d'abord, j'examine le problème du Déséquilibre de Classe (DC) dans le cadre de l'apprentissage supervisé en introduisant un modèle analytiquement solvable basé sur la mécanique statistique: je propose un cadre théorique pour analyser et interpréter le problème de DC. Certains phénomènes non triviaux sont observés : par exemple, un ensemble d'entraînement équilibré aboutit souvent à une performance sous-optimale. Ensuite, j'étudie le phénomène de blocage dynamique dans les verres structuraux à l'aide de modèles avancés d'AA. En exploitant des réseaux de neurones sur graphe qui sont SE(3)-équivariants, j'atteins des performance qui atteignent ou surpassent l'état de l'art pour la prédiction des propriétés dynamiques à partir de la structure statique. Cela suggère l'émergence d'un "ordre amorphe" qui est corrélé avec la dynamique. Cela souligne également l'importance des features directionnelles dans l'identification de cet ordre. Ensemble, ces contributions démontrent le potentiel de la physique statistique pour résoudre les défis de l'AA et l'utilité des modèles d'AA pour faire progresser les sciences physiques.

Title: Statistical Physics - Machine Learning Interplay: from Addressing Class Imbalance with Replica Theory to Predicting Dynamical Heterogeneities with SE(3)-equivariant Graph Neural Networks

Keywords: Machine Learning, Class Imbalance, Replica Trick, Supercooled Liquids, Steerable Convolutions, Graph Neural Networks

Abstract: This thesis explores the relationship between Machine Learning (ML) and Statistical Physics (SP), addressing two significant challenges at the interface between the two fields. First, I examine the problem of Class Imbalance (CI) in the supervised learning set-up by introducing an analytically tractable model grounded in statistical mechanics: I provide a theoretical framework to analyze and interpret CI. Some non-trivial phenomena are observed: for example, a balanced training set often results in sub-optimal performance. Second, I study the phenomenon of dynamical arrest in supercooled liquids through advanced ML models. Leveraging SE(3)-equivariant Graph Neural Networks, I am able to reach or surpass state-of-the art accuracy in the task of prediction of dynamical properties from static structure. This suggests the emergence of a growing "amorphous order" that correlates with particle dynamics. It also emphasizes the importance of directional features in identifying this order. Together, these contributions demonstrate the potential of SP in addressing ML challenges and the utility of ML models in advancing physical sciences.

Résumé en Français

Dans cette thèse, j'explore l'interaction entre l'Apprentissage Automatique et la Physique Statistique, en mettant en lumière leur relation mutuellement bénéfique. D'une part, l'apprentissage automatique peut servir d'outil puissant pour identifier des motifs complexes dans les données, aidant ainsi les physiciens à développer des théories. D'autre part, les méthodes analytiques développées par la communauté de la physique statistique pour étudier des systèmes complexes avec de nombreux degrés de liberté peuvent être appliquées à l'étude théorique des modèles d'apprentissage automatique, offrant des perspectives précieuses sur leurs mécanismes sous-jacents.

Dans le premier chapitre, j'examine une direction de cette relation en abordant un défi récurrent dans l'apprentissage automatique: le problème du déséquilibre des classes, à l'aide d'un modèle analytiquement solvable. Un problème central dans les approches d'apprentissage automatique réside dans le fait que les ensembles de données représentent souvent les différentes classes de manière déséquilibrée, en raison des méthodes de collecte des données ou des caractéristiques intrinsèques de la tâche étudiée. Cela constitue un problème significatif, car apprendre à partir de données déséquilibrées peut produire des résultats trompeurs, les modèles reflétant cet déséquilibre et pouvant conduire les utilisateurs à des conclusions erronées. Bien que le déséquilibre des classes soit bien reconnu dans la communauté de l'apprentissage automatique et ait été traité par diverses approches empiriques, ces méthodes manquent souvent de garanties et il est difficile de choisir la plus efficace dans différents contextes. Cette incertitude provient en grande partie de l'absence d'une théorie unifiée du déséquilibre des classes. Ces dernières années, des physiciens ont abordé cette problématique en développant des modèles théoriques pour expliquer le phénomène et proposer des stratégies théoriquement fondées pour atténuer le déséquilibre des classes. Dans ce chapitre, je contribue à cette ligne de recherche en étudiant un modèle analytiquement tractable, le *Teacher-Student Perceptron*, à l'aide des outils de la physique statistique. En m'appuyant sur ce cadre paradigmatique, j'ai modélisé les données d'entrée pour reproduire un déséquilibre des classes du type détection d'anomalies, où le déséquilibre est intrinsèque au problème plutôt qu'un résultat du processus de collecte des données. Ce cadre permet de clarifier le rôle du déséquilibre des classes dans l'entraînement (provenant de la collecte des données) par rapport au déséquilibre des classes intrinsèque. De plus, ce cadre offre une interprétation claire de l'impact du déséquilibre, distinguant les "bons" modèles des "mauvais" et identifiant les facteurs clés grâce à une argumentation basée sur le compromis entre énergie et entropie. En même temps, ce cadre valide la fiabilité de plusieurs métriques de performance couramment utilisées dans des contextes empiriques, en identifiant la *Balanced Accuracy* comme la métrique la plus efficace. De manière intéressante, ce modèle simple révèle un comportement très non

trivial: contrairement à la pratique courante, un ensemble d'entrînement constituée de données équilibrées conduit souvent à une performance sous-optimale.

Dans le second chapitre, j'explore la direction inverse de cette relation interdisciplinaire en étudiant le phénomène de blocage dynamique dans les verres structuraux à l'aide de modèles d'apprentissage profond. Les verres structuraux sont des matériaux amorphes, c'est-à-dire des matériaux sans aucun type d'ordre cristallin. Dans ces matériaux, les fonctions de corrélation à deux points simples, couramment utilisées en physique pour détecter un ordre structurel, ne parviennent pas à identifier un arrangement régulier à longue portée des particules. Un aspect particulièrement intrigant des verres structuraux est la présence d'une transition dynamique: en dessous d'une certaine température, ces matériaux se comportent comme des solides, bien qu'aucun changement visible ne se produise dans l'arrangement structurel des particules. Cela contraste fortement avec la cristallisation traditionnelle des solides, où les particules s'alignent en un réseau régulier lors de la transition de phase. Les physiciens se demandent depuis longtemps s'il existe un "ordre amorphe" dans la structure des verres, une forme d'organisation qui pourrait croître à mesure que la température diminue et expliquer le ralentissement observé de la dynamique (la transition vers la solidité). Cette question est loin d'être triviale et reste un défi ouvert dans l'étude des matériaux désordonnés. Pour tenter de répondre à cette question, les physiciens se tournent vers l'apprentissage automatique, déclenchant une nouvelle ligne de recherche dans laquelle des modèles d'apprentissage automatique sont employés pour identifier des descripteurs structurels complexes susceptibles de capturer un ordre amorphe sous-jacent et de le corréler au ralentissement de la dynamique. Je contribue à cette recherche en utilisant un modèle avancé d'apprentissage automatique, les réseaux neuronaux SE(3)-équivariants. Inspirés par la théorie des groupes, ces modèles disposent de bases théoriques solides, garantissant que les *features* apprises respectent la symétrie roto-translationnelle inhérente au matériaux. J'ai pu prédire le champ de mobilité avec une grande précision à partir de la structure statique des particules dans des simulations numériques des verres. Dans ces matériaux, le champ de mobilité présente une hétérogénéité spatiale, caractérisée par des régions distinctes "lentes" et "rapides". Mon modèle a réussi à capturer les corrélations spatiales de ce champ et a démontré une transférabilité en température, indiquant qu'il avait efficacement appris une représentation robuste de la structure statique. Cette représentation suggère l'existence d'un ordre amorphe qui croît à mesure que la température diminue et correspond à l'échelle de longueur croissante des observables dynamiques, telles que la taille typique des domaines "rapides" et "lents". Bien que le modèle lui-même ne soit pas entièrement interprétable, son accent sur les informations directionnelles souligne l'importance d'incorporer et de combiner des caractéristiques vectorielles des empilements atomiques, au-delà des simples caractéristiques invariantes par rotation, pour identifier l'ordre amorphe.

Les deux problématiques examinées dans ce manuscrit représentent seulement quelques exemples des nombreuses instances de fertilisation croisée entre ces domaines. J'espère que ce travail mettra en avant la valeur de cette relation, en encourageant une collaboration accrue dans les années à venir. Une telle synergie promet une compréhension plus approfondie des outils avancés d'apprentissage automatique, tout en faisant progresser notre compréhension des phénomènes physiques.

Contents

Acknowledgements

I would like to thank my équipe de direction for their guidance throughout these years, and in particular my supervisor, François, whose support and understanding have been invaluable. He granted me great freedom in my research while remaining closely involved, always demonstrating genuine care for my future and well-being.

I also thank my collaborators, Marco Baity Jesi and Valentina Ros, for being a source of inspiration and positivity. Their insights opened new perspectives on the research world, and their warmth and encouragement made me feel welcomed and valued.

I thank the jury, and in particular the rapporteurs, for taking the time to review my manuscript and for offering valuable insights to refine it and inspire future research. I also appreciate their presence at my defense, especially given its proximity to the holidays.

I want to thank all my friends from the A-team, an incredible group. It's thanks to you that I now consider Paris a little bit like home. We have grown together over these years and have learned to support each other, each with our own quirks. We've realized that even though life isn't one big party, we'll never skip the final round.

I thank my friends from the Circolo letterario. You are all people I have always deeply admired, and it has been an honor for me to be part of this group. We shared not only our journey but also our efforts, worries, and a good dose of laughs and reels. Hold on, guys, Switzerland is close!

I thank the Riders on the Trove, my roots. You can take the Riders out of Pilla, but you can't take Pilla out of the Riders.

I thank Dezzu, Gerry, Rob, and all the friends with whom I've shared my passion for music, which has kept me going all these years.

Thanks to my girlfriend for being there at the end of this journey and for shaking up my life. Thank you for showing me that a smile is more important than a deadline, and that a cup of coffee can sometimes be better than a fresh beer.

Finally, thanks to my family. You are the most genuine example of unconditional love. Thank you for showing me every day what it means to give without expecting anything in return. Growing up with your example has been a blessing, and every day I strive to be even just a little bit like you.

Introduction

Today, discussing the successes of Artificial Intelligence (AI), particularly Machine Learning (ML), and why they're compelling fields of study might seem redundant. Over the past two decades, advances in data availability, GPU-powered parallel computing, and open-source frameworks have made implementing ML more accessible than ever. Coupled with a capitalistic drive to over-perform and automate, machine learning has permeated our society to such an extent that even this manuscript could plausibly have been generated by a language model, without raising suspicion among non-experts. Regardless of whether this deep integration of AI is beneficial or concerning, studying AI remains essential for understanding our era. It fosters an awareness of the inherent limitations and biases in these systems, shaped largely by the privileged few who wield the power to influence their design.

The rise of ML has revolutionized numerous aspects of society, with scientific disciplines, and particularly physics, being no exception. The capacity of ML models to detect complex patterns within data has proven a valuable tool for scientists. Indeed, the adoption of ML approaches across various branches of physics has expanded rapidly, with applications ranging from materials science [CM17, KMB$^+$16, CYZ$^+$19] to high-energy physics [PdON18, AAA$^+$19], molecular dynamics simulations [RPK17, CSMT18], ab-initio calculations [HSD$^+$21] and numerous others not cited here [CCC$^+$19].

An intriguing aspect of the relationship between machine learning and physics is that, while the application of ML techniques to physics tasks is relatively recent, the communities studying these fields have a deep-rooted connection. In particular, the statistical physics community made significant contributions to early theoretical studies on machine learning models. This synergy became especially prominent from the late 1980s, with the development of the Hopfield model [Hop82][1], an artificial network grounded in spin glass concepts that could function as an associative memory system, and Elizabeth Gardner's foundational studies on the learning capacity of the perceptron, the simplest neural network [GD89]. During this period, and especially in the early 1990s, researchers in disordered physics produced numerous stud-

[1] Hopfield was awarded the Nobel Prize in Physics in 2024 *"for foundational discoveries and inventions that enable machine learning with artificial neural networks"*. More information can be found at `https://www.nobelprize.org/uploads/2024/09/advanced-physicsprize2024.pdf`

ies employing spin glass techniques to explore the theoretical behavior of basic neural networks [Gyo90, SST92, Eng12].

Since those early years, the machine learning community has made remarkable advances in practical applications, largely by scaling up models: with vastly more clean data available, models have grown significantly in the number of adjustable parameters. These advances were driven by the development of deep feed-forward networks and increasingly effective algorithms for training them. Remarkably, deep learning achieved a major leap by revisiting the same architectures proposed in the 1980s and 1990s, applying them on a much larger scale in terms of both data and parameters [KSH12, LBH15, HZRS15, GPAM$^+$14, VSP$^+$17]. However, these deep architectures are challenging to analyze using statistical mechanics (SM) approaches. Despite this, physicists continue to study neural networks through SM tools, as many phenomena observed in deep nonlinear networks can be mimicked in simpler models that are easier to analyze [dRBK20, dSB21, GSd$^+$19, GAS$^+$19, MMN18, DKL$^+$23, CKZ23].

With a growing number of numerical experiments revealing intriguing phenomenology in neural networks, much of which remains unexplained, a growing community of machine learners suggest that the task of interpreting these results falls also to physicists, who can apply tools specialized to study the emergence of collective behavior from the interaction of a large number of degrees of freedom to better understand the behavior of these complex models [Zde20].

In my thesis, I explore the interface between these two disciplines, focusing on their mutually beneficial relationship and how they can foster each other's development. On the one hand, machine learning can serve as a powerful tool for identifying complex patterns in data, helping physicists in developing theories by highlighting key quantities and phenomena to investigate. On the other hand, the analytical methods developed by the statistical physics community to study complex systems with many interacting degrees of freedom can be applied to the theoretical study of machine learning models, providing valuable insights into their underlying mechanisms.

In the first chapter, I examine one direction of this relationship by addressing a prevalent challenge in machine learning, the issue of class imbalance, using an analytically tractable model. As discussed earlier, the success of machine learning depends, among other factors, on the availability of large amounts of data. However, a central problem is that these datasets often represent different classes in an unbalanced way, stemming from the methods of data collection or the inherent characteristics of the task under study. This is a significant issue because learning from unbalanced data can yield misleading results, with models reflecting this imbalance and potentially leading users to incorrect conclusions. While class imbalance is well-recognized in the machine learning community and has been tackled with various empirical approaches, these methods often lack guarantees and it is challenging to choose the more effective in different contexts. This uncertainty largely stems from the absence of a unified theory of class imbalance. In recent years, physicists have approached this issue by developing theoretical models to explain the phenomenon and propose theoretically grounded strategies to mitigate class imbalance. In this chapter, I contribute to this line of research by proposing an analytically tractable model that aims at capturing the phenomenology of anomaly detection problems. This model, being both exactly solvable and interpretable, enables us to identify the underlying causes and main mechanisms driving the problem. Although I do not claim to have fully elucidated the complexities of real-world networks and tasks, this toy model offers insight into simpler cases

and provides guidance on potentially relevant aspects to investigate in more complex scenarios. This approach underscores a key advantage of statistical physics methods: while they may not capture the full intricacies of large models, they can effectively isolate different mechanisms at play, guiding more advanced analyses.

In the second chapter, I explore the reverse direction of this interdisciplinary relationship by examining the phenomenon of dynamical arrest in supercooled liquids using advanced machine learning models. Supercooled liquids are amorphous materials *i.e.* materials that lack a long range crystalline order. Many common materials are amorphous, including window glass (silica oxide), basalt (and various other rocks), most plastics (complex polymer assemblies that rarely crystallize), as well as pastes, gels, and creams. In these materials, simple two-point correlation functions, commonly used in physics to detect structural order, fail to identify any long-range regular arrangement of particles. One particularly intriguing aspect of supercooled liquids is the presence of a dynamical transition: below a certain temperature, these materials behave like solids, even though no visible change occurs in the structural arrangement of particles. This stands in stark contrast to traditional crystallization in solids, where particles align into a regular lattice upon phase transition. As I'll explore further, physicists have long questioned whether an "amorphous order" exists within the structure of supercooled liquids, a form of organization that might grow as temperature decreases and account for the observed slowdown in dynamics (the transition to solidity). This question is far from trivial and remains an open challenge in the study of disordered materials. In pursuit of this answer, physicists have turned to machine learning, sparking a new line of research in which ML models are employed to identify complex structural descriptors that might capture an underlying amorphous order and correlate it with the slowdown in dynamics. I contribute to this research by employing an advanced machine learning model, Roto-Translational Equivariant Neural Networks. Inspired by group theory, these models have robust theoretical foundations, ensuring that the features learned respect the roto-translational symmetry inherent to the material. I adapted this architecture to handle large assemblies of molecules in supercooled liquids and achieved state-of-the-art results. While these findings do not fully elucidate the nature of amorphous order, they strongly suggest its existence and indicate that directional features are essential for identifying it.

The two problems examined in this manuscript represent just a few examples of the many instances of cross-fertilization between these fields. Hopefully, this work will underline the value of this relationship, encouraging further collaboration in the years ahead. Such synergy promises a deeper understanding of advanced ML tools, while also advancing our understanding of physical phenomena. In essence, this partnership has the potential to enhance our understanding of the world, and both disciplines stand to gain from embracing each other's insights.

Chapter 1

Class Imbalance in Exactly Solvable Models

Contents

In this chapter, we investigate the impact of class imbalance on learned models using statistical mechanics tools. We begin by reviewing the SM framework and demonstrating how it applies to learning problems, translating physical concepts into machine learning terminology. Next, we introduce key techniques from the field of glassy physics that are relevant to our analysis. Following this, we address the problem of class imbalance, discussing its effects and examining the main empirical strategies developed by the machine learning community to mitigate these effects. We then delve into the theoretical foundations underlying our approach. Finally, we

discuss our theoretical setup, analysis, and results on anomaly-detection class imbalance for perceptron learning, highlighting our ability to uncover non-trivial insights in a relatively simple model. Some of these insights are further supported by experimental findings and existing literature.

1.1 Statistical Mechanics of Learning

Statistical Mechanics, being concerned with the collective behavior of systems composed of a large number of interacting components, such as molecules or atoms, offers a powerful set of tools and methodologies for addressing complex problems in Machine Learning. The fundamental premise of Statistical Mechanics is to describe how macroscopic properties emerge from microscopic interactions, a perspective that aligns well with the challenges in Machine Learning, where the goal is to understand and optimize the behavior of models composed of numerous interacting parameters. In this context, the *parameters* of machine learning model can be viewed as the *degrees of freedom*, analogous to position and velocity of particles in a physical system. The techniques developed in statistical mechanics, particularly mean-field approaches for spin glass models, provide valuable insights into the *typical* behavior of these parameters and allow to extract *macroscopic* quantities like the generalization performances of the model from the *microscopic* interaction of the large number of parameters. This approach represents a paradigm shift in the theoretical study of machine learning models by focusing on typical behavior, rather than on the statistical worst-case bounds that represent a common focus in statistical learning theory. Building on the seminal works of Gardner on the storage capacity of neural networks [Gar87] and of Györgyi and Tishby on learning a perceptron rule [Gyo90], a prolific body of research has emerged [SST92, FM93, CDS23, CMV$^+$23, MKL$^+$20, LPCM24], continuing to the present day, and demonstrating the effectiveness of statistical mechanics methods in addressing these types of problems.

1.1.1 Gibbs Learning, typical behavior

Supervised Classification is a fundamental task in modern machine learning. This problem is also referred to as *learning from examples* since a model has to infer a labelling rule from the observation of a set of input-output relationships. Following the notation of Seung *et al.* in [SST92], it can be defined in the following way: we have a set of examples, or training data-points, $\{\mathbf{S}^\ell\}_{\ell=1}^P$ with $\mathbf{S}^\ell = (S_1^\ell, \dots, S_M^\ell)$ M-dimensional vector and P number of data points. A label or class membership, is associated to each sample through a labelling rule $\sigma_0(\mathbf{S}^\ell)$, thus the training dataset is composed of P input-output pairs $(\mathbf{S}^\ell, \sigma_0(\mathbf{S}^\ell))$. A model σ with a set of N parameters $\mathbf{w} = (w_1, \dots, w_N)$ is defined by the input-output relation: $\sigma^\ell = \sigma(\mathbf{w}; \mathbf{S}^\ell)$ and has to adjust its weights during the training procedure in order to best reproduce the original labeling rule σ_0. The training procedure is performed through the minimization of the training loss (or energy in the SM dictionary):

$$E(\mathbf{w}) = \sum_{\ell=1}^{P} \epsilon(\mathbf{w}; \mathbf{S}^\ell) \tag{1.1}$$

where the error function $\epsilon(\mathbf{w};\mathbf{S}^\ell)$ is a measure of the distance between the network prediction and the true labeling rule. A popular choice is the quadratic error function:

$$\epsilon(\mathbf{w};\mathbf{S}^\ell) = \frac{1}{2}[\sigma(\mathbf{w};\mathbf{S}^\ell) - \sigma_0(\mathbf{S}^\ell)]^2 \tag{1.2}$$

Learning from examples can be framed as an *optimization problem* where the goal is to minimize the training loss to find the optimal parameters $\mathbf{w}^* = \text{argmin}_{\mathbf{w}} E(\mathbf{w})$. This optimization can be performed using methods such as gradient descent (GD):

$$\frac{\partial \mathbf{w}}{\partial t} = -\nabla_{\mathbf{w}} E(\mathbf{w}) \tag{1.3}$$

This approach ensures that the training loss decreases, leading to good performance on the training set, which consists of a limited number of samples. However, it does not guarantee good generalization, meaning it does not ensure that the model correctly learns the original labeling rule. To evaluate generalization, one should examine the generalization error, which measures the performance of the learned model on new, unseen samples. Assuming a certain population distribution $\mathrm{d}\mu(\mathbf{S})$, this is quantified as follows:

$$\epsilon_g = \langle\!\langle \epsilon(\mathbf{w}^*;\mathbf{S}) \rangle\!\rangle_{\mathrm{d}\mu} \tag{1.4}$$

where the symbol $\langle\!\langle \ldots \rangle\!\rangle_{\mathrm{d}\mu}$ denotes the average over the distribution of the sample. This set-up is very general and, to apply it to a specific problem, one must specify the main building blocks: the data distribution $\mathrm{d}\mu$, which is often unavailable in practice and must be approximated for analytical computations; the labeling rule σ_0 which denotes the relationship between a sample and its class and the model σ which is parametrized by learnable weights and whose architecture needs to be fixed by choosing the input-output relation. Furthermore one needs to specify the form of the energy function. We presented the common choice of the L2-loss, but many alternatives are available (hinge loss, cross-entropy loss, *etc.*).

Traditional GD dynamics have notable limitations in both training efficiency and generalization performance. Specifically, GD can become trapped in local minima or saddle points, which slows the learning process, but also risks converging to sharp minima that correspond to solutions that do not generalize well [KMN+17, WZE17]. To address these issues, stochastic optimization techniques such as Stochastic Gradient Descent (SGD) were developed [Bot99]. In this study, we explore an alternative approach known as Langevin dynamics, defined as follows:

$$\frac{\partial \mathbf{w}}{\partial t} = -\nabla_{\mathbf{w}} E(\mathbf{w}) + \boldsymbol{\eta}(t) \tag{1.5}$$

where $\boldsymbol{\eta}(t)$ is a white noise with variance:

$$\langle \eta_i(t)\eta_j(t') \rangle = 2T\delta_{ij}\delta(t-t') \tag{1.6}$$

the parameter T measures the amplitude of the stochastic noise and we will refer to it as *temperature* in the SM vocabulary. According to Statistical Mechanics, an ergodic system evolving under Langevin dynamics will reach an equilibrium state after a sufficiently long period [Gar09].

At equilibrium, the system samples different configurations according to the Gibbs probability distribution:

$$P(\mathbf{w}) = \frac{1}{Z} e^{-\beta E(\mathbf{w})} \tag{1.7}$$

with $\beta = 1/T$. In our framework, the *equilibrium* state corresponds to the *end of the training regime*. Once the model has been trained for sufficiently long according to the dynamics described by Eq. 1.5, each solution $\mathbf{w}^*$ is associated with a probability $P(\mathbf{w}^*)$. By knowing this probability distribution, we can extract meaningful insights into the typical behavior of the learned model by computing statistical averages. To achieve this, we need to take the *thermodynamic limit*, where $N \to \infty$, which in machine learning terms corresponds to the *limit of infinite number of parameters*: in this limit the behavior concentrates around the typical one and non-typical cases become negligible thus statistical averages coincide with typical behavior (in non-pathological thermodynamic phases). Here we also define two quantities that will be fundamental in the following:

$$Z = \int \mathrm{d}\mathbf{w} e^{-\beta E(\mathbf{w})} \tag{1.8}$$

$$F = -\frac{1}{\beta} \log Z \tag{1.9}$$

Z is called *partition function* and corresponds to the normalization factor in Eq. 1.7. F is called *free energy* and contains all the information about the equilibrium state of a system.

Up to this point, we have assumed a fixed realization of the training data. However, in practice, training data is sampled from an underlying distribution $\mathrm{d}\mu_{\text{train}}$ and then held fixed throughout training. To compute expectations properly, one must also average over the initial distribution of samples. This is analogous to what occurs in Spin Glass models with quenched disorder. The statistical mechanics community has developed numerous tools to study such systems, which we will introduce in the next sections.

1.1.2 Spin Glass models and quenched disorder

The connection between statistical mechanics and machine learning becomes especially pronounced when considering spin glass models, a type of disordered system characterized by the presence of quenched disorder. While a comprehensive treatment of the physics underlying spin glasses lies beyond the scope of this manuscript, we will discuss key concepts relevant to our derivation. For an introductory discussion on Spin Glass Physics, see [CC05]; for a thoroughly referenced review, see *e.g.* [ABJ24].

Quenched disorder Disorder refers to any form of irregularity or randomness in the structure of the system's components. Quenched disorder can be thought of as "frozen" randomness. Imagine a material where some of its components, like impurities or defects, are randomly distributed but do not change over time. These imperfections are introduced during the process of creating or cooling the material, and once set, they remain fixed. This is what we refer to as quenched disorder: it reflects randomness that is static and does not evolve as the system evolves. Quenched disorder can be found in various systems, including magnetic materials,

alloys, and glasses [Myd93]. For example, in some alloys, certain atoms may be replaced by magnetic impurities in random positions, creating quenched disorder. This fixed randomness can have significant effects on the system's macroscopic behavior, such as its ability to conduct electricity or respond to external magnetic fields. As we'll detail in the following, in the context of machine learning, the role of disorder is played by the training samples. These samples are randomly drawn from an underlying distribution and remain fixed throughout the learning process

Self-averageness Each sample of a material has its own peculiar realization of the quenched disorder that is determined by the way in which the sample was prepared. Thus melting and instantaneously cooling-down multiple times a single sample produces different realizations of the disorder *i.e.* the defects freeze in different positions each time. It is impractical to study each specific instance of disorder in detail. Instead, physicists introduce a probability distribution to describe the disorder, allowing to focus on the typical behavior of the system. Fortunately, many generic properties of these systems do not depend on the specific realisation of the quenched disorder, or more precisely these properties average out when considering a larger sample size, and are called *self-averaging*. For these quantities, the average over the disorder distribution coincides with the typical behavior when considering the thermodynamic limit:

$$A^{typ} \xrightarrow{N\to\infty} \langle\!\langle A \rangle\!\rangle_{\mu_{disorder}} \tag{1.10}$$

One of such quantities is the free energy F introduced in Eq. 1.9.

An example To illustrate the concepts introduced in the previous paragraphs, let us examine a paradigmatic spin-glass model, the Edwards-Anderson (EA) model [EA75]. As introduced above, spin-glasses are alloys in which magnetic impurities substitute the original atoms in positions randomly selected during the chemical preparation of the sample. This results in magnetic impurities distributed in space in a disordered fashion that interact between them through a magnetic coupling. This complex system is effectively mimicked by the EA model where impurities are arranged in a regular fashion on a cubic lattice, each impurity is described by a state variable s_i that can assume two values $s_i \in \{-1, +1\}$ and the disorder in the space arrangement is converted into disorder in the interactions, meaning that two impurities interact between them with an interaction J_{ij} that is random. The energy of such model reads:

$$E(\mathbf{s}) = -\sum_{\langle ij \rangle} J_{ij} s_i s_j \tag{1.11}$$

with J_{ij} independent and identically distributed according to a gaussian distribution and $\langle ij \rangle$ denotes the sum over nearest neighbors in the cubic lattice. Notice that the state of the system $\mathbf{s} = (s_1, \ldots, s_N)$ here plays the same role as the network weights $\mathbf{w}$ in Eq. 1.1 and the random interactions J_{ij} correspond to the training samples $\mathbf{S}^\ell$. To characterize the model, one should compute the typical free energy which, by leveraging the *self-averaging* property, can be obtained through averaging over the disorder distribution. Since, for the models we are interested in, the free energy is an *extensive* quantity i.e. it scales with the size of the system N, we focus on its density to have a finite result:

$$f^{typ} = -\lim_{N\to\infty} \frac{1}{N\beta} \langle\!\langle logZ \rangle\!\rangle_{\mathrm{d}\mu(\{J_{ij}\})} = -\lim_{N\to\infty} \frac{1}{N\beta} \int \prod_{ij} \mathrm{d}J_{ij} P(J_{ij}) \log \int \mathrm{d}\mathbf{s} e^{-\beta E(\mathbf{s})} \tag{1.12}$$

Going back to the *learning from examples* set-up, to extract the typical behavior of the learned model averaged over the distribution of input samples one should compute:

$$f^{typ} = -\lim_{N\to\infty} \frac{1}{N\beta} \int \prod_{\ell} \mathrm{d}\mathbf{S}^{\ell} P(\{\mathbf{S}^{\ell}\}) \log \int \mathrm{d}\mathbf{w} e^{-\beta E(\mathbf{w})} \tag{1.13}$$

where $P(\{\mathbf{S}^{\ell}\})$ is the joint distribution of training samples. In the simplest scenario these are assumed to be i.i.d. and the distribution factorizes. Physicists have developed various techniques to compute or estimate the analytical form of the quenched free energy of disordered systems. In particular, we will focus on the replica method, which plays a central role in the analysis presented in this work.

1.1.3 The replica method

The replica method allows *exact* evaluation of the quenched free energy [MPV87]. Although it is a *non-rigorous* technique, its results have been shown to align with those obtained through more mathematically rigorous methods [BM19, Shi18, Pas22]. The core idea behind the method is the smart use of the identity:

$$\log Z = \lim_{n\to 0} \frac{Z^n - 1}{n} \tag{1.14}$$

As already introduced above, the main goal is the evaluation of the quenched free energy introduced in Eq. 1.13:

$$-\beta f^{typ} = \lim_{N\to\infty} \frac{1}{N} \langle\!\langle \log Z \rangle\!\rangle_{\mu_{train}} \tag{1.15}$$

with $\langle\!\langle \ldots \rangle\!\rangle_{\mu_{train}} = \int \prod_{\ell} \mathrm{d}\mathbf{S}^{\ell} P(\{\mathbf{S}^{\ell}\})$. Computing the quenched average of the logarithm of the partition function is a hard problem, but exploiting the relation of Eq. 1.14, one can rewrite the quenched free energy as:

$$-\beta f^{typ} = \lim_{N\to\infty} \lim_{n\to 0} \frac{1}{Nn} \log \langle\!\langle Z^n \rangle\!\rangle_{\mu_{train}} \tag{1.16}$$

For the identity to hold rigorously, n should be a real number, but in that case there would be no advantage at all in computing the r.h.s. compared to the l.h.s. However, if one promotes n to be an integer, Z^n can be written as the product of partition functions of n replicas of the same system:

$$-\beta f^{typ} = \lim_{N\to\infty} \lim_{n\to 0} \frac{1}{Nn} \log \langle\!\langle \int \mathrm{d}\mathbf{w}_1 \, \mathrm{d}\mathbf{w}_2 \ldots \mathrm{d}\mathbf{w}_n e^{-\beta E(\mathbf{w}_1) - \beta E(\mathbf{w}_2) \cdots - \beta E(\mathbf{w}_n)} \rangle\!\rangle_{\mu_{train}} \tag{1.17}$$

All the replicas here experience the same disorder, this makes the analytical treatment much easier, as we will show in the full computation of the quenched free energy in next sections. Once the average is computed, one should perform the analytical continuation of Eq. 1.17 to $n \to 0$ and take the limit. This step is crucial to the replica method and needs to be done with care as the analytical continuation may not be unique [DV01, VZ85]. At this stage, it's worth mentioning that a common practice in replica calculations is to switch the thermodynamic limit and the $n \to 0$ limit to allow the saddle-point calculation. In addition to ensuring uniqueness, this approach requires a stability check of the retrieved stationary points. In our work we adopt a model known to avoid such instabilities [SST92] so we do not perform this check directly. For an insightful study on saddle-point stability in spin glass models, see [dAT78].

1.2 Class Imbalance

Supervised learning has become a cornerstone of modern machine learning, underpinning a wide range of applications from image classification to natural language processing. Central to this paradigm is the availability of labeled data, where the learning algorithm infers patterns from a set of input-output pairs. A fundamental assumption in supervised learning is that the training data represents the underlying distribution of the target domain. However, **in many real-world scenarios**, this assumption is violated due to the presence of class imbalance, where **one or more classes are significantly underrepresented in comparison to others** [YTWM00, AHY11, KHM+21, SGPC+23].

Class imbalance poses a serious challenge to supervised learning models. When the distribution of classes in the training data is skewed, **machine learning algorithms tend to become biased towards the majority class**. This bias can lead to poor predictive performance on the minority class, which is often of greater interest in practical applications. For instance, in medical diagnostics, detecting rare diseases (minority class) is crucial, yet models may struggle to identify these cases due to the overwhelming number of healthy instances (majority class).

The consequences of class imbalance extend beyond reduced accuracy on minority classes. Imbalanced datasets can cause misleading evaluation metrics, such as accuracy, to overestimate model performance by favoring high-quality predictions for the majority class. This can mask the true deficiencies of the model, particularly in scenarios where the minority class is of critical importance. As a result, addressing class imbalance is not merely a matter of improving overall model accuracy, but of ensuring fair and effective decision-making in contexts where minority classes carry significant weight.

Different kinds of imbalance Imbalance in data can arise from various factors. Depending on the underlying cause, class imbalance can be categorized into two types: *intrinsic* and *extrinsic*. *Intrinsic imbalance* occurs when the natural distribution of classes in the real world is inherently imbalanced. This reflects the true nature of the data, such as in medical diagnosis where certain diseases are rare compared to healthy cases. On the other hand, *extrinsic imbalance* arises due to external factors, such as limitations in data collection processes, where some classes are underrepresented due to practical constraints rather than their actual frequency. **We introduce a further distinction by classifying two types of imbalance based on the shape of the distribution of training data**. *Outlier or Anomaly Detection (AD) imbalance*, is generally a binary problem. All examples are drawn from the same distribution, and one needs to identify outliers based on an unknown rule (*e.g.* only some of the components of a power grid will cause a failure, but we do not know what will cause it [ZSZ+19]). In this case, the imbalance is intrinsic to the problem, as anomalies are naturally fewer in number than normal samples. In contrast, *Multiple Groups (MG) imbalance* involves samples drawn from distinctly different distributions, with the imbalance arising either from the sampling process (*e.g.* the toxicity of certain chemicals is tested more often than others [SGPC+23]) or being intrinsic to the data itself (*e.g.* some species being more common within an ecosystem [KHM+21]). As we will see, a feature that makes MG imbalance different from AD imbalance is that, differently from MG, AD imbalance has an associated intrinsic imbalance scale, which we will call ρ_0 and represents the fraction of anomalies. If $\rho_0 = 0.5$, common data and anomalies appear with the

same frequency.

Performance Metrics In the context of supervised classification, performance metrics used to assess a model's effectiveness are computed starting from the entries of the confusion matrix. Each row of the matrix represents the instances in an actual class while each column represents the instances in a predicted class. In particular, when considering a *binary classification* set-up with two classes referred to as *Positives* and *Negatives* the confusion-matrix takes the following form:

		Predicted Class	
		Positive	Negative
Actual Class	Positives	True Positives (TP)	False Negatives (FN)
	Negatives	False Positives (FP)	True Negatives (TN)

(1.18)

The most common metric is the *accuracy* (a), defined as the ratio of correctly predicted samples to the total number of observed samples:

$$a = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}} \tag{1.19}$$

It is highly sensitive to class imbalance and it can provide misleading results as it can be biased towards the majority class. Indeed **a "dummy" model that predicts always the majority class in an highly imbalanced problem will have an high accuracy even though it's not able to predict at all the minority class**. To better assess model performance on imbalanced datasets, several alternative metrics are commonly used. In order to define them, we introduce the *Recall* (r) and the *Specificity* (s) which correspond respectively to the True Positive Rate and the True Negative Rate:

$$r = \frac{TP}{TP + FN} \tag{1.20}$$

$$s = \frac{TN}{TN + FP} \tag{1.21}$$

The most used metrics in imbalanced problems are the *Balanced Accuracy* (a_{bal}), *Precision* (p) and *F1-Score* (F_1):

- *Balanced Accuracy* is computed as the average between recall and specificity. This metric weights the accuracy on the majority and the minority class in the same way regardless of the amount of imbalance:

$$a_{\mathrm{bal}} = \frac{(r + s)}{2} \tag{1.22}$$

- *Precision* measures the proportion of correctly predicted positive instances among all instances predicted as positive:

$$p = \frac{TP}{TP + FP} \tag{1.23}$$

- *F1-score* is obtained as the harmonic mean of precision and recall. For this reason a high F1-score implies both high precision and recall while when one of the two quantities is low also the F1-score deteriorates. This metric is able to locate a model that is able to predict a large number of positive samples with high precision.

$$F_1 = 2\frac{p \cdot r}{p + r} \tag{1.24}$$

1.2.1 Mitigation strategies

Addressing the detrimental effects of class imbalance has led to the development of multiple approaches, with the machine learning community establishing common rules of thumb based on empirical tests. These approaches can be broadly categorized into three types: those acting on the data distribution, those modifying the loss function, and those biasing the dynamics of the training process. In the following we review some representative works of the three categories.

Data Distribution Standard methods consist in random under/over sampling (RUS and ROS): the goal of these methods is to re-balance the training set by acting on the sampling procedure. In ROS the training set is sampled with replacement from the original data-set [JS02]. In RUS, random samples from the majority class are discarded. More involved over-sampling techniques consist in the generation of synthetic examples for the minority class either in the original data-space [CBHK02] or in the deep feature space of a Neural Network [AH17]. Advanced under-sampling techniques consist in downsizing the majority class by selection of *near miss* or *most distant* examples *i.e.* majority samples that are closest to, or farthest from, the minority samples [ZM03]. Many works comparing multiple re-sampling methods are present in literature: Kamalov *et al.* compare different re-sampling methods to establish best sampling ratio with extensive experiments on multiple datasets. They use shallow learning models (SVM and RF). They conclude that rarely full-resampling (i.e. exactly balancing the train set) produces the best balanced accuracy or F1-score [KAE22]. The main drawbacks of these methods are that under-sampling techniques discard data, thus they're not well suited for problems where few data is available while over-sampling techniques can lead to overfitting of the minority class.

Objective function First approaches in this direction consisted in re-weighting the loss of each sample based on their class [XM89]. The goal of such approaches is to assign a larger weight in the loss function to examples from minority classes and force a better accuracy on them. Latest works focus on more refined re-parametrization of the loss that are effective also in over-parametrized regimes [KPOT21] [BKVT23] [TKVB22] [MJR+21]. These works offer also theoretical guarantees based mainly on the Unconstrained Features Model (UFM) which is a model that mimics deep networks but it's analytically treatable. It is important to note that while ROS is equivalent to simple loss reweighting when considering Gradient Descent dynamics or the infinite-data limit, in practical scenarios where algorithms like SGD are employed, these two approaches exhibit distinct behavior.

Learning algorithm These methods aim to bias the learning dynamics to suppress the domination of the majority class. Tang *et al.* exploit a causal framework to dynamically modify the momentum of the SGD algorithm [THZ20]. Francazi *et al.* modify the SGD algorithm

by computing gradients separately on the minority and majority class and re-scaling them by their norm. This helps counter the directional noise introduced by imbalance in the training dynamics [FBJL23].

1.2.2 Theoretical studies

Due to the **lack of a theoretical framework** for the analysis of Class Imbalance, **it is often unclear which mitigation strategy works best and when or why**. For this reason, recent studies tried to fill this theoretical gap, either by focusing on how imbalance influences the learning dynamics, or on how it influences the loss landscape. We review some representative works that **investigate the statistical properties of the loss landscape under class imbalance**: we introduce the main building blocks and features of their analytical set-up that will serve as foundations for our analysis in the subsequent sections.

Teacher-Student Perceptron The Perceptron model is a paradigmatic model for theoretical machine learning studies. Despite its simplicity it offers an exact analytical framework, making it valuable for exploring fundamental concepts. The model captures key aspects of the behavior seen in more complex systems, while remaining accessible enough to allow for intuitive understanding and precise mathematical treatment. Since the seminal work of Gardner and Derrida [GD89] it has been widely studied in all its variants [Gyo90, SST92, FM93, Nis01] and has become a cornerstone in understanding the statistical mechanics of learning. These studies have explored its phase transitions, generalization capabilities, and error landscapes. The Perceptron is a linear model characterized by a set of $N+1$ learnable parameters: weights $\mathbf{w} = (w_1, \dots, w_N)$ and a bias b. Given an input sample $\mathbf{S}^\ell = (S_1^\ell, \dots, S_N^\ell)$, its input-output relation reads as:

$$g^\ell = g\left(\frac{\mathbf{w}\cdot\mathbf{S}^\ell}{\sqrt{N}} + b\right), \tag{1.25}$$

where the factor $1/\sqrt{N}$ is there to ensure that the scalar product is $O(1)$ as the bias and g is a non-linear activation function. Multiple variants of the Perceptron model have been developed, primarily differing in the choice of weight domains (either real-valued or discrete) and the form of the activation function. These modifications significantly influence the model's learning dynamics and capacity. For instance, real-valued weights allow for finer adjustments and smoother optimization landscapes, while discrete weights may lead to simpler representations but more rugged energy landscapes. Similarly, the choice of activation function, whether it is a step function, a sigmoid, or a ReLU, affects the model's ability to handle non-linearities, convergence properties, and generalization performance.

The most common framework for studying the Perceptron model is the *teacher-student* setup. In this set-up, a *teacher* perceptron with a planted weight configuration assigns a label to each sample, while a *student* perceptron learns to mimic the teacher by adjusting its weights through Empirical Risk Minimization on the samples in the training set. This approach allows to rigorously analyze how well the student model generalizes from finite training data, how efficiently it can approximate the teacher's function, and how the complexity of the teacher influences learning difficulty. Ground-truth labels g_0^ℓ are obtained through the teacher assignment:

$$g_0^\ell = g\left(\frac{\mathbf{w}^0\cdot\mathbf{S}^\ell}{\sqrt{N}} + b_0\right). \tag{1.26}$$

Although teacher and student have the same architecture, the teacher's parameters $\mathbf{w}^0$ and b_0 are fixed, while the student parameters $\mathbf{w}$ and b are learned through the minimization of the loss

$$\mathcal{E}(\mathbf{w}, b) = \sum_{\ell=1}^{P} \epsilon \left(g\left(\frac{\mathbf{w} \cdot \mathbf{S}^\ell}{\sqrt{N}} + b \right), g_0^\ell \right), \tag{1.27}$$

The choice of the specific form of the error function ϵ influences as well the behavior of the model under training. Multiple reviews are available where different combinations are investigated [SST92, WRB93, Eng12].

Perceptron and Class Imbalance *Mignacco et al.* [MKL+20] investigate the effects of class imbalance on a perceptron trained on a Gaussian Mixture. The model used to generate the data is a mixture of 2 gaussian clusters in N dimensions. Each data point $\mathbf{S}^\ell$ is described by:

$$\mathbf{S}^\ell = \frac{\mathbf{v}^*}{\sqrt{d}} y^\ell + \sqrt{\Delta}\mathbf{z}^\ell \tag{1.28}$$

where $y_\ell \in \{+1, -1\}$ denotes the class of the samples which correspond to the cluster membership (one cluster contains positives and the other negatives). $\mathbf{z}_i$ and $\mathbf{v}^*$ are $\sim \mathcal{N}(0, 1)$. They introduce a parameter ρ that determines the fraction of positive samples in the train-set: denoting with P the number of training samples, ρP have label $y = +1$ and $(1 - \rho)P$ have label $y = -1$. Learning is performed through Empirical Risk Minimization with the loss:

$$\mathcal{E}(\mathbf{w}, b) = \sum_{\ell=1}^{P} \epsilon \left(\frac{\mathbf{w} \cdot \mathbf{S}^\ell}{\sqrt{N}} + b, y^\ell \right) + \frac{1}{2}\lambda ||\mathbf{w}||_2^2 \tag{1.29}$$

where the activation is linear and a regularization term is introduced with respect to (1.27). They perform an exact analytical analysis though the use of Gordon Inequalities and investigate thoroughly the effect of the regularization term and different choices of the loss-function ϵ (square, logistic and hinge). In particular they observe that *imbalance i.e.* $\rho \neq 0.5$ deteriorates the quality of learning and impedes achieving Bayes optimal performances.

Mannelli et al. [MGRS23] introduce a more involved model for generating data: the teacher-mixture. Data is sampled from two symmetric gaussians: $\mathbf{S} \sim \mathcal{N}(\pm\mathbf{v}/\sqrt{N}, \Delta_\pm \mathbb{I}^{N\times N})$ each one with probability ρ and $1 - \rho$. For each group a teacher perceptron determines the labels with a planted configuration $\mathbf{W}_T^+$ and $\mathbf{W}_T^-$. The loss is computed taking into account that for each sub-population there's a different teacher:

$$\mathcal{E}(\mathbf{w}, b) = \sum_{\ell=1}^{P} \epsilon \left(g\left(\frac{\mathbf{w} \cdot \mathbf{S}^\ell}{\sqrt{N}} + b \right), g\left(\frac{\mathbf{W}_T^{c_\ell} \cdot \mathbf{S}^\ell}{\sqrt{N}} + b_T^{c_\ell} \right) \right) + \frac{1}{2}\lambda ||\mathbf{w}||_2^2 \tag{1.30}$$

the choice for the loss-function ϵ is the cross-entropy loss. They perform exact computations via the Replica Method in the $T \to 0$ limit which corresponds to the absence of randomness in the dynamics. They focus mostly on fairness implications, studying how imbalance affects the performances across the sub-populations. They also introduce a mitigation strategy based on coupled neural networks trained on subsets of the full training dataset.

Loffredo et al. [LPCM24] analyze imbalance in a set-up where the data is discrete. Each data point is composed of L (the data dimension) categorical variables that can assume Q values. In

our notation: $\mathbf{S}^\ell \in \{0,1\}^{L\times Q}$ assuming one-hot encoding for the categorical variables. Similarly to the gaussian-mixture set-up, they assign labels based on cluster membership, thus two sub-populations are present in the data each one associated to one class. They don't do strong assumption on the shape of the distribution but fix the first two moments of the two sub-populations. They study a linear model which represents a generalisation of the perceptron to categorical data:

$$g^\ell = \mathrm{sgn}\left[\sum_{ik} \frac{W_{ik}S_{ik}}{\sqrt{L}} - b\right] \tag{1.31}$$

where $\mathbf{W} \in \mathbb{R}^{L\times Q}$ and $b \in \mathbb{R}$ are the learned parameters. The learning is performed trough Empirical Risk Minimization on the hinge loss. They investigate the effect of imbalance on various accuracy metrics and in particular show that the AUC score is rather insensitive to imbalance while the Balanced Accuracy is a better suited metrics to study imbalanced problems. They focus also on the effectiveness of re-sampling techniques and prove that mixed strategies of random over-sampling/under-sampling are the most effective.

Limitations All the **recent analytical works** presented do not make the distinction between AD and MG imbalance that we introduced in Sec. 1.2. Most often they **implicitly address MG imbalance relying on the assumption of two distinct sub-populations** in the data distribution. Our work differs from the latter in its focus on a different kind of imbalance, namely Anomaly Detection, where features in both classes are instances of an overarching distribution. One consequence is that some of the re-balancing solutions analyzed in [LPCM24] do not apply in our context.

1.3 Anomaly-Detection Class Imbalance for perceptron learning

We study the effects of AD imbalance on the training and test landscape in a paradigmatic analytically tractable model. Specifically, we study a modified version of the Teacher-Student (TS) spherical perceptron [SST92], where one can tune the amount of class imbalance, and study its effect on learning. Studying a tractable model, where the ground truth is known, allows us to disentangle the various reasons why a high performance is reached or not, providing interpretable results. We investigate the theoretical trends of various performance metrics and compare our predictions with experiments in realistic settings.

1.3.1 Model

We consider the widely-studied Teacher-Student Spherical Perceptron, a variant of the model introduced in Sec. 1.2.2 where the parameters are real numbers, namely:

$$\mathbf{w} = (w_1, \dots, w_N) \in \mathbb{R}^N,\ b \in \mathbb{R} \tag{1.32}$$

and weights are subject to the *spherical constraint*:

$$\mathbf{w}^T \cdot \mathbf{w} = N \tag{1.33}$$

this choice fixes the norm of the weight vector and it acts similarly to the *regularization* term in the loss (1.29). The activation function is $g(x) = \text{sign}(x)$ thus the output of the network is discrete: $g^\ell \in \{+1, -1\}$.

The train-set is composed of $P = N\alpha$ samples $\mathbf{S}^\ell \in \mathbb{R}^N$ ($\ell = 1, \dots, P$ is the sample index). The number of train samples is chosen to scale with the data/weights dimension in order to have an extensive energy (1.27) in the thermodynamic limit. The parameter $\alpha \sim O(1)$ is called *data-scarcity* parameter and determines the training regime:

$\alpha > 1$	under-parametrized
$\alpha < 1$	over-parametrized

This parameter can have a big impact on the behavior of the model: TS-perceptrons with discrete weights show phase transition to perfect learning when increasing α above a certain critical value α^* [SST92]. Our model with continuous weights doesn't show any discontinuities when varying the data-scarcity parameter.

As introduced in Sec. 1.2.2 ground-truth labels g_0^ℓ are obtained through the teacher assignment which has the same architecture of the student and whose parameters $(\mathbf{w_0}, b)$ are fixed. Learning is performed through minimization of the loss (1.27) where the loss function is the square-loss $\epsilon(x, y) = \frac{1}{2}(x - y)^2$.

Modeling Class Imbalance The novelty of our computation consists in modeling the AD Imbalance by introducing an imbalance parameter ρ, that fixes the ratio between samples in the majority and minority classes.

Samples are i.i.d., and for each sample $\mathbf{S}$ the components are distributed according to: $S_i \sim DS_i = \frac{\mathrm{d}S_i}{\sqrt{2\pi}} e^{-S_i^2/2}$. We will use the shorthand notation $D\mathbf{S} = \prod_{i=1}^N DS_i$ to denote the measure of probability of the single sample.

In order to have a skewed distribution of samples with a number $N\alpha\rho_{\text{train}}$ of positive samples ($g_0^\ell = +1$) and $N\alpha(1-\rho_{\text{train}})$ of negative samples ($g_0^\ell = -1$), we define the training set measure as:

$$\mathrm{d}\mu_{\text{train}}(\{\mathbf{S}^\ell\}) = \frac{1}{c_+^{N\alpha\rho_{\text{train}}}}\left(\prod_{\ell=1}^{\alpha N\rho_{\text{train}}} D\mathbf{S}^\ell\,\Theta\left(\frac{\mathbf{w}^0\cdot\mathbf{S}^\ell}{\sqrt{N}}+b_0\right)\right)\times \\ \times\frac{1}{c_-^{N\alpha(1-\rho_{\text{train}})}}\left(\prod_{\ell'=\alpha N\rho_{\text{train}}+1}^{\alpha N} D\mathbf{S}^{\ell'}\,\Theta\left(-\frac{\mathbf{w}^0\cdot\mathbf{S}^{\ell'}}{\sqrt{N}}-b_0\right)\right) \tag{1.34}$$

The notation $\mathrm{d}\mu_{\text{train}}(\{\mathbf{S}^\ell\})$ is shorthand for $\mathrm{d}\mu_{\text{train}}(\mathbf{S}^1,\dots,\mathbf{S}^{N\alpha})$ and denotes the measure of probability over all the training samples. We use the Heaviside (Θ) function to select the samples according to the relevant output sign of the Teacher Perceptron. $c_+ = 1/2\,\mathrm{Erfc}(-b_0/\sqrt{2})$ and $c_- = 1-c_+$ represent respectively the normalization constant for positive and negative samples. Note that $\mathrm{d}\mu_{\text{train}}$ is in general not a Gaussian measure, since in the direction of $\mathbf{w}^0$, it is a piecewise Gaussian with normalization factors which depend on ρ_{train}.

Train, Population and Test Imbalance Here, we elaborate on the definition and roles of the imbalance ratio ρ for training set, population and test set.

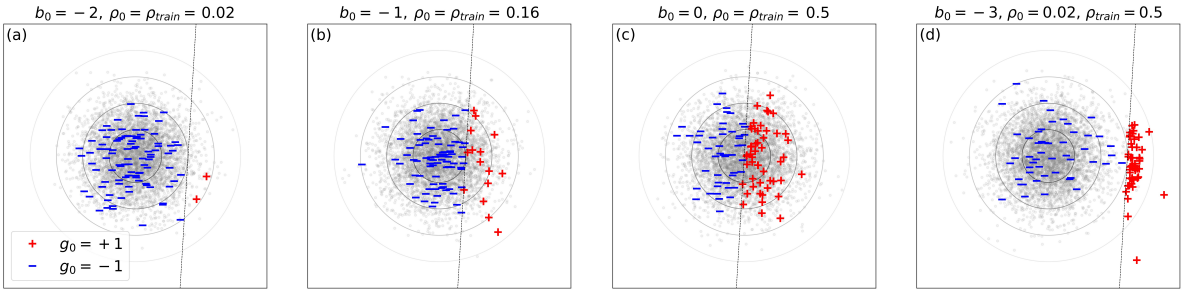


Figure 1.1: *(a,b,c)* **Intrinsic Imbalance**. 2D sketches showing the effect of the teacher bias b_0 on the intrinsic imbalance. Normal examples (negative label, $g_0^\ell = -1$) are represented with blue $-$ symbols, anomalies (positive label, $g_0^\ell = +1$) with red $+$. Shaded grey points depict the underlying Gaussian data distribution and grey circles locate contours at $1\sigma, 2\sigma$ and 3σ (σ is the standard deviation). The black dashed line represents the teacher decision boundary which determines the ground-truth labels. All the examples are extracted from the underlying Gaussian distribution and no specific imbalance ratio is imposed externally. Increasing the magnitude of the teacher bias b_0 ((c) $\to$ (a)) translates the teacher boundary (black dashed line) away from the origin cutting the tails of the Gaussian distribution, making anomalies more rare. *(c,d)* **Informative samples**. Two cases are compared where the training imbalance is fixed to $\rho_{\text{train}} = 0.5$ while the teacher biased is varied. As $|b_0|$ grows, anomalies become more and more concentrated around the decision boundary, becoming more informative about the teacher's direction.

The introduction of ρ_{train} in Eq. (1.34) allows to control the amount of imbalance in the **training set**. This parameter is externally imposed and enables us to explore scenarios where the model is trained with varying levels of imbalance.

When ρ_{train} is not fixed, *i.e.* when samples are extracted from the Gaussian measure $\mathrm{d}\mu_{\text{pop}}(\{\mathbf{S}^\ell\}) \propto \prod_{\ell=1}^{\alpha N} D\mathbf{S}^\ell$, an imbalance also arises naturally due to the presence of the bias parameter b_0 in the

Teacher model. It can be computed as:

$$\rho_0(b_0) = P\left(\frac{\mathbf{w}_0 \cdot \mathbf{S}}{\sqrt{N}} + b_0 > 0\right) = \frac{1}{2}\mathrm{Erfc}\left(-\frac{b_0}{\sqrt{2}}\right). \tag{1.35}$$

We refer to this as **Intrinsic Imbalance** as it measures the intrinsic imbalance present in the data generation process. It depends solely on the teacher's bias, and describes the rarity of observing anomalies. It can be easily visualized geometrically: when dealing with the population, all the samples are drawn from a Gaussian distribution centered in the origin, while varying b_0 amounts to translating the Teacher decision boundary away from the origin of the N-dimensional space. Thus, one of the two classes lies on the tail of the distribution and is less represented. This is depicted in 2D sketches of Fig. 1.1 (a,b,c), where a decreasing b_0 (in magnitude) results in a less biased teacher and a less imbalanced population.

Since the population distribution is rarely available in practice, a **test set** consisting of samples not observed during training is introduced in standard machine learning settings to test the performance of the trained model. We define the test distribution as:

$$\mathrm{d}\mu_{\mathrm{test}}(\mathbf{S}) = \frac{\rho_{\mathrm{test}}}{c_+}\Theta\left(\frac{\mathbf{w}^0 \cdot \mathbf{S}}{\sqrt{N}} + b_0\right) D\mathbf{S} + \frac{1-\rho_{\mathrm{test}}}{c_-}\Theta\left(-\frac{\mathbf{w}^0 \cdot \mathbf{S}}{\sqrt{N}} - b_0\right) D\mathbf{S} \tag{1.36}$$

where ρ_{test} measures the probability of having an anomaly in the test set. Common choices are $\rho_{\mathrm{test}} = 0.5$ (balanced test set) or $\rho_{\mathrm{test}} = \rho_0(b_0)$ (test set that reflects the intrinsic imbalance). As we will discuss below, while some performance metrics do not explicitly depend on ρ_{test}, others do. The choice of the test imbalance is as important as the train imbalance, and can lead to misleading results if not properly considered.

Informative samples The teacher bias b_0 also determines how informative the two classes are. When $b_0 \neq 0$, the samples with label sign opposite to that of the bias (class 1 if $b_0 < 0$ or class -1 if $b_0 > 0$ – in other words, the anomaly, or minority class) concentrate on the decision boundary of the teacher, while the samples of the other class have a lower probability to lie on the boundary. Thus the samples of the minority class (in the population) are more informative about the decision boundary (*i.e.* the direction of the teacher hyperplane). Fig. 1.1 (c,d) shows this effect through 2-dimensional sketches. The consequences of this feature are particularly relevant when the student has information about the teacher's bias. This specific scenario is discussed in Sec. 1.3.3. It is possible to evaluate the probability of a sample lying on the boundary and showing this effect quantitatively: let's consider a sample $\mathbf{S}$ extracted from an N-dimensional gaussian centered in the origin:

$$P(\mathbf{S}) = \frac{1}{(2\pi)^{N/2}} e^{-1/2 \sum_{i=1}^{N} S_i^2} \tag{1.37}$$

We're interested in the case with $b_0 < 0$ (the positives are the anomalies) and we want to compute the probability that the sample lies between the teacher decision boundary and the parallel hyperplane at a distance δx such that $g_0 > 0$, namely:

$$P\left(0 < \frac{\mathbf{w}_0 \cdot \mathbf{S}}{\sqrt{N}} + b_0 < \delta x \,\middle|\, \frac{\mathbf{w}_0 \cdot \mathbf{S}}{\sqrt{N}} + b_0 > 0\right) = \frac{P\left(0 < \frac{\mathbf{w}_0 \cdot \mathbf{S}}{\sqrt{N}} + b_0 < \delta x\right)}{P\left(\frac{\mathbf{w}_0 \cdot \mathbf{S}}{\sqrt{N}} + b_0 > 0\right)} \tag{1.38}$$

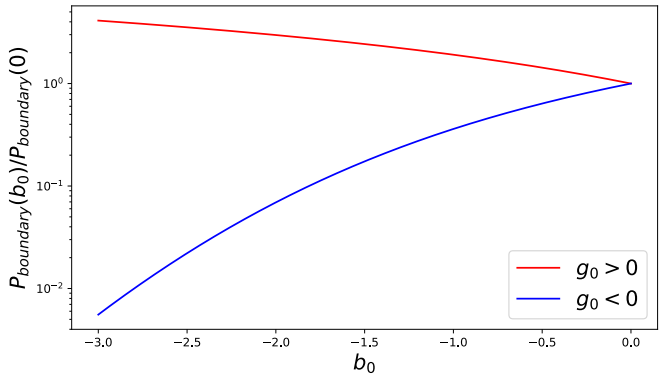


Figure 1.2: **Probability of being close to the boundary**: for the minority ($g_0 > 0$) and for the majority class ($g_0 < 0$). As a function of b_0, we plot the quantities $P^-_{boundary}(b_0)$ and $P^+_{boundary}(b_0)$, divided by their value at $b_0 = 0$.

We can now evaluate the denominator:

$$P\left(\frac{\mathbf{w}_0\cdot\mathbf{S}}{\sqrt{N}}+b_0>0\right)=\int D\mathbf{S}\Theta\left(\frac{\mathbf{w}_0\cdot\mathbf{S}}{\sqrt{N}}+b_0\right) \tag{1.39}$$

$$=\int \mathrm{d}y\Theta(y+b_0)\int D\mathbf{S}\delta\left(y-\frac{\mathbf{w}_0\cdot\mathbf{S}}{\sqrt{N}}\right) \tag{1.40}$$

By exploiting the Fourier representation of the δ one gets:

$$P\left(\frac{\mathbf{w}_0\cdot\mathbf{S}}{\sqrt{N}}+b_0>0\right)=\int \mathrm{d}y\Theta(y+b_0)\int\frac{\mathrm{d}\hat{y}}{2\pi}e^{i\hat{y}y}\int D\mathbf{S}e^{-i\hat{y}\frac{\mathbf{w}_0\cdot\mathbf{S}}{\sqrt{N}}} \tag{1.41}$$

$$=\int \mathrm{d}y\Theta(y+b_0)\int\frac{\mathrm{d}\hat{y}}{2\pi}e^{i\hat{y}y-1/2\hat{y}^2} \tag{1.42}$$

$$=\int\Theta(y+b_0)Dy=\frac{1}{2}\mathrm{Erfc}\left(\frac{-b_0}{\sqrt{2}}\right) \tag{1.43}$$

As for the numerator, the computation follows the same lines and one gets:

$$P\left(0<\frac{\mathbf{w}_0\cdot\mathbf{S}}{\sqrt{N}}+b_0<\delta x\right)=\int_{-b_0}^{-b_0+\delta x}Dy\xrightarrow{\delta x\to 0}\frac{1}{\sqrt{2\pi}}e^{-b_0^2/2}\delta x \tag{1.44}$$

Similarly, one can compute the probability for samples with $g_0 < 0$ and finally get:

$$P^+_{boundary}=\frac{\frac{1}{\sqrt{2\pi}}e^{-b_0^2/2}\delta x}{\frac{1}{2}\mathrm{Erfc}\left(\frac{-b_0}{\sqrt{2}}\right)} \tag{1.45}$$

$$P^-_{boundary}=\frac{\frac{1}{\sqrt{2\pi}}e^{-b_0^2/2}\delta x}{1-\frac{1}{2}\mathrm{Erfc}\left(\frac{-b_0}{\sqrt{2}}\right)} \tag{1.46}$$

The two quantities are shown in Fig.1.2 versus $b_0 < 0$. It is clear that positive samples have a higher probability to lie on the boundary as long as the teacher bias is negative, so they represent the minority class in the population distribution.

1.3.2 Theoretical Analysis

Here we perform the detailed calculation of the free energy density of our model in the thermodynamic limit and show how to extract relevant features and performances of the learned model.

Replica Calculation

As introduced in Sec. 1.1.3 the goal of the replica calculation is the evaluation of the free energy density in the thermodynamic limit, namely:

$$-\beta f^{typ} = \lim_{N\to\infty}\lim_{n\to 0}\frac{1}{Nn}\log\langle\!\langle Z^n\rangle\!\rangle_{\mu_{train}} \tag{1.47}$$

the form of μ_{train} is the one of Eq. 1.34. To make the notation lighter, we abbreviate ρ_{train} as ρ for the whole derivation, since we are now only focusing on the training set. We start by evaluating the replicated partition function:

$$\langle\!\langle Z^n\rangle\!\rangle = \int \mathrm{d}\mu_{\text{train}}(\{\mathbf{S}^\ell\})\int\left(\prod_{\sigma=1}^{n}\mathrm{d}\mu(b_\sigma)\,\mathrm{d}\mu(\mathbf{w}^\sigma)\right)e^{-\beta\sum_\sigma\sum_\ell\epsilon(\mathbf{w}^\sigma,b_\sigma;\mathbf{S}^\ell)} \tag{1.48}$$

where we have used the shorthand notation for the loss $\mathcal{E}(\mathbf{w},b)=\sum_{\ell=1}^{N\alpha}\epsilon(\mathbf{w},b;\mathbf{S}^\ell)$ to denote the dependence of term ℓ in the sum on the student weights and on the sample ℓ. $\mathrm{d}\mu(\mathbf{w})$ denotes the integration measure over the student weights and it enforces the spherical constraint:

$$\mathrm{d}\mu(\mathbf{w}) = \prod_{i=1}^{N}\frac{\mathrm{d}w_i}{\sqrt{2\pi e}}\delta(\mathbf{w}\cdot\mathbf{w}-N) \tag{1.49}$$

$\mathrm{d}\mu(b)=\mathrm{d}b$ is the integration measure over the student bias.

By expanding the integration measure over the training samples and collecting respectively positive and negative terms we get:

$$\begin{aligned}\langle\!\langle Z^n\rangle\!\rangle = \int\prod_{\sigma=1}^{n}\mathrm{d}\mu(b_\sigma)\,\mathrm{d}\mu(\mathbf{w}^\sigma)\left[\frac{1}{c_+}\int D\mathbf{S}\Theta\left(\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}+b_0\right)e^{-\beta\sum_\sigma\epsilon(\mathbf{w}^\sigma,b_\sigma;\mathbf{S})}\right]^{N\alpha\rho}\cdot\\ \cdot\left[\frac{1}{c_-}\int D\mathbf{S}\Theta\left(-\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}-b_0\right)e^{-\beta\sum_\sigma\epsilon(\mathbf{w}^\sigma,b_\sigma;\mathbf{S})}\right]^{N\alpha(1-\rho)}\end{aligned} \tag{1.50}$$

We define:

$$\mathcal{G}_r^{\pm}(\{\mathbf{w}^\sigma,b_\sigma\}) = -\log\frac{1}{c_\pm}\int D\mathbf{S}\Theta\left(\pm\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}\pm b_0\right)e^{-\beta\sum_\sigma\epsilon(\mathbf{w}^\sigma,b_\sigma;\mathbf{S})} \tag{1.51}$$

where $\{\mathbf{w}^\sigma,b_\sigma\}$ denotes the dependence of $\mathcal{G}_r^{\pm}$ on the whole set of n replicated students. Then one can rewrite the replicated partition function as:

$$\langle\!\langle Z^n\rangle\!\rangle = \int\prod_{\sigma=1}^{n}\mathrm{d}\mu(b_\sigma)\,\mathrm{d}\mu(\mathbf{w}^\sigma)e^{-N\alpha\rho\mathcal{G}_r^{+}(\{\mathbf{w}^\sigma,b_\sigma\})-N\alpha(1-\rho)\mathcal{G}_r^{-}(\{\mathbf{w}^\sigma,b_\sigma\})} \tag{1.52}$$

We introduce the auxiliary variables x_σ and y in order to remove $\mathbf{S}$ from the argument of g function in the expression of $\mathcal{G}_r^{\pm}$:

$$e^{-\mathcal{G}_r^+} = \frac{1}{c_+}\int\prod_{\sigma=1}^{n}\mathrm{d}x_\sigma\int\mathrm{d}y\,\Theta(y+b_0)e^{-\frac{\beta}{2}\sum_\sigma[g(x_\sigma+b_\sigma)-g(y+b_0)]^2}\int D\mathbf{S}\prod_{\sigma=1}^{n}\delta\left(x_\sigma-\frac{\mathbf{w}^\sigma\cdot\mathbf{S}}{\sqrt{N}}\right)\delta\left(y-\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}\right) \tag{1.53}$$

$$= \frac{1}{c_+}\int\prod_{\sigma=1}^{n}\frac{\mathrm{d}x_\sigma\,\mathrm{d}\hat{x}_\sigma}{2\pi}\int\frac{\mathrm{d}y\,\mathrm{d}\hat{y}}{2\pi}\Theta(y+b_0)e^{\left(-\frac{\beta}{2}\sum_\sigma[g(x_\sigma+b_\sigma)-g(y+b_0)]^2+i\sum_\sigma x_\sigma\hat{x}_\sigma+iy\hat{y}\right)}\times$$
$$\times\int D\mathbf{S}e^{-i\left(\sum_\sigma\mathbf{w}^\sigma\hat{x}_\sigma+\mathbf{w}^0\hat{y}\right)\cdot\frac{\mathbf{S}}{\sqrt{N}}} \tag{1.54}$$

where we exploited the Fourier representation of the δ-function and introduced the conjugate variables $\hat{x}_\sigma$ and $\hat{y}$. The last integral is Gaussian and it yields: $e^{-\frac{1}{2N}\left(\sum_\sigma\mathbf{w}^\sigma\hat{x}_\sigma+\mathbf{w}^0\hat{y}\right)^2}$. The function $\mathcal{G}_r^+$ depends on the vectors $\mathbf{w}^\sigma,\mathbf{w}^0$ only through the overlaps:

$$Q_{\sigma\sigma'} = \frac{\mathbf{w}^\sigma\cdot\mathbf{w}^{\sigma'}}{N},\quad R_\sigma = \frac{\mathbf{w}^\sigma\cdot\mathbf{w}^0}{N}, \tag{1.55}$$

which are emergent order parameters of the theory. They measure respectively the alignment between replicated students and between replicas and the teacher. In terms of these functions, $\mathcal{G}_r^+$ can be written as:

$$e^{-\mathcal{G}_r^+} = \frac{1}{c_+}\int\prod_{\sigma=1}^{n}\frac{\mathrm{d}x_\sigma\,\mathrm{d}\hat{x}_\sigma}{2\pi}\int\frac{\mathrm{d}y\,\mathrm{d}\hat{y}}{2\pi}\Theta(y+b_0)e^{-\frac{\beta}{2}\sum_\sigma[g(x_\sigma+b_\sigma)-g(y+b_0)]^2+i\sum_\sigma x_\sigma\hat{x}_\sigma+iy\hat{y}}\times$$
$$\times e^{-\frac{1}{2}\sum_{\sigma,\sigma'}\hat{x}_\sigma\hat{x}_{\sigma'}Q_{\sigma\sigma'}-\hat{y}\sum_\sigma\hat{x}_\sigma R_\sigma-\frac{1}{2}\hat{y}^2}, \tag{1.56}$$

and similarly

$$e^{-\mathcal{G}_r^-} = \frac{1}{c_-}\int\prod_{\sigma=1}^{n}\frac{\mathrm{d}x_\sigma\,\mathrm{d}\hat{x}_\sigma}{2\pi}\int\frac{\mathrm{d}y\,\mathrm{d}\hat{y}}{2\pi}\Theta(-y-b_0)e^{-\frac{\beta}{2}\sum_\sigma[g(x_\sigma+b_\sigma)-g(y+b_0)]^2+i\sum_\sigma x_\sigma\hat{x}_\sigma+iy\hat{y}}\times$$
$$\times e^{-\frac{1}{2}\sum_{\sigma,\sigma'}\hat{x}_\sigma\hat{x}_{\sigma'}Q_{\sigma\sigma'}-\hat{y}\sum_\sigma\hat{x}_\sigma R_\sigma-\frac{1}{2}\hat{y}^2}. \tag{1.57}$$

The replicated partition function can thus be written as an integral over the order parameters:

$$\langle\!\langle Z^n\rangle\!\rangle = \int\prod_{\sigma=1}^{n}\mathrm{d}\mu(b_\sigma)\,\mathrm{d}\mu(\mathbf{w}^\sigma)e^{-N\alpha\rho\mathcal{G}_r^+(\{\mathbf{w}^\sigma,b_\sigma\})-N\alpha(1-\rho)\mathcal{G}_r^-(\{\mathbf{w}^\sigma,b_\sigma\})}\times$$
$$\times\int\prod_{\sigma>\sigma'}\mathrm{d}Q_{\sigma\sigma'}\int\prod_\sigma\mathrm{d}R\sigma\prod_{\sigma>\sigma'}\delta(\mathbf{w}^\sigma\cdot\mathbf{w}^{\sigma'}-NQ_{\sigma\sigma'})\prod_\sigma\delta(\mathbf{w}^\sigma\cdot\mathbf{w}^0-NR_\sigma) \tag{1.58}$$

$$= \int\prod_\sigma\mathrm{d}b_\sigma\int\prod_{\sigma>\sigma'}\frac{\mathrm{d}Q_{\sigma\sigma'}\,\mathrm{d}\hat{Q}_{\sigma\sigma'}}{2\pi i}\int\prod_\sigma\frac{\mathrm{d}R_\sigma\,\mathrm{d}\hat{R}_\sigma}{2\pi i}e^{-N\alpha\rho\mathcal{G}_r^+(\{Q_{\sigma\sigma'},R_\sigma,b_\sigma\})-N\alpha(1-\rho)\mathcal{G}_r^-(\{Q_{\sigma\sigma'},R_\sigma,b_\sigma\})}\times$$
$$\times e^{N\left(-\sum_{\sigma>\sigma'}Q_{\sigma\sigma'}\hat{Q}_{\sigma\sigma'}-\sum_\sigma\hat{R}_\sigma R_\sigma\right)}\int\prod_{\sigma=1}^{n}\mathrm{d}\mu(\mathbf{w}^\sigma)e^{\sum_{\sigma>\sigma'}\hat{Q}_{\sigma\sigma'}\mathbf{w}^\sigma\mathbf{w}^{\sigma'}+\sum_\sigma\hat{R}_\sigma\mathbf{w}^\sigma\mathbf{w}^0} \tag{1.59}$$

with $\hat{Q}_{\sigma\sigma'}$ and $\hat{R}_\sigma$ conjugates of the overlaps. Then:

$$\langle\!\langle Z^n\rangle\!\rangle = \int\prod_\sigma\mathrm{d}b_\sigma\int\prod_{\sigma>\sigma'}\frac{\mathrm{d}Q_{\sigma\sigma'}\,\mathrm{d}\hat{Q}_{\sigma\sigma'}}{2\pi i}\int\prod_\sigma\frac{\mathrm{d}R_\sigma\,\mathrm{d}\hat{R}_\sigma}{2\pi i}e^{-N\mathcal{A}_r(\{Q_{\sigma\sigma'},\hat{Q}_{\sigma\sigma'},R_\sigma,\hat{R}_\sigma\})} \tag{1.60}$$

with:

$$\mathcal{A}_r(\{Q_{\sigma\sigma'}, \hat{Q}_{\sigma\sigma'}, R_\sigma, \hat{R}_\sigma\}) = \alpha\rho\mathcal{G}_r^+(\{Q_{\sigma\sigma'}, R_\sigma, b_\sigma\}) + \alpha(1-\rho)\mathcal{G}_r^-(\{Q_{\sigma\sigma'}, R_\sigma, b_\sigma\}) - \mathcal{G}_0(\{Q_{\sigma\sigma'}, \hat{Q}_{\sigma\sigma'}, R_\sigma, \hat{R}_\sigma\}) \tag{1.61}$$

and

$$\mathcal{G}_0(\{Q_{\sigma\sigma'}, \hat{Q}_{\sigma\sigma'}, R_\sigma, \hat{R}_\sigma\}) = -\sum_{\sigma>\sigma'} Q_{\sigma\sigma'}\hat{Q}_{\sigma\sigma'} - \sum_\sigma \hat{R}_\sigma R_\sigma + \frac{1}{N}\log\int\prod_{\sigma=1}^{n} \mathrm{d}\mu(\mathbf{w}^\sigma) e^{\sum_{\sigma>\sigma'}\hat{Q}_{\sigma\sigma'}\mathbf{w}^\sigma\mathbf{w}^{\sigma'} + \sum_\sigma \hat{R}_\sigma \mathbf{w}^\sigma \mathbf{w}^0}. \tag{1.62}$$

Replica Symmetric Ansatz In order to evaluate the integral in (1.60) by saddle-point we switch the order of the two limits, getting:

$$-\beta f^{typ} = \lim_{n\to 0}\frac{1}{n}\min_{\substack{Q_{\sigma\sigma'}, R_\sigma, \\ \hat{Q}_{\sigma\sigma'}, \hat{R}_\sigma, b_\sigma}} \left\{-\mathcal{A}_r(\{Q_{\sigma\sigma'}, \hat{Q}_{\sigma\sigma'}, R_\sigma, \hat{R}_\sigma\})\right\}. \tag{1.63}$$

To carry on the computation one has to find a parametrization of the order parameters and express (1.63) as a function of the elements of these multi-dimensional arrays and the number of replicas n. We stick to the Replica Symmetric (RS) case where one assumes that the replicated students are symmetric i.e. they all have the same overlap with the teacher and among them:

$$Q_{\sigma\sigma'} = \delta_{\sigma\sigma'} + (1-\delta_{\sigma\sigma'})q \tag{1.64}$$
$$\hat{Q}_{\sigma\sigma'} = \delta_{\sigma\sigma'} + (1-\delta_{\sigma\sigma'})\hat{q} \tag{1.65}$$
$$R_\sigma = R \tag{1.66}$$
$$\hat{R}_\sigma = \hat{R} \tag{1.67}$$
$$b_\sigma = b. \tag{1.68}$$

We stress that in our computation the student bias b is treated as an order parameter and its value is fixed by saddle point as for the other parameters. We define:

$$G_r^\pm = \lim_{n\to 0}\frac{\mathcal{G}_r^\pm}{n}, \qquad G_0 = \lim_{n\to 0}\frac{\mathcal{G}_0}{n} \tag{1.69}$$

then the optimization problem to solve in order to find the equilibrium values of the order parameters becomes the following:

$$-\beta f^{typ} = \min_{\substack{q, R, \\ \hat{q}, \hat{R}, b}} \left\{G_0(q, \hat{q}, R, \hat{R}) - \alpha\rho G_r^+(q, R, b) - \alpha(1-\rho)G_r^-(q, R, b)\right\} \tag{1.70}$$

We'll refer to $G_r^\pm$ as the energetic terms as they represent the energetic contribution to the free energy of positive and negative class samples. G_0 is the entropic or volume term and quantifies the number of student configurations that correspond to a given choice of the order parameters.

In the following we show the detailed computations for G_r^+, the derivation follows the same lines for G_r^-. The plan is to substitute the RS ansatz in the expression of $\mathcal{G}_r^+$ and integrate over the conjugates variables $\hat{y}, \hat{x}_\sigma$: the integral in $\hat{y}$ is a Gaussian integral and yields:

$$e^{-\mathcal{G}_r^+} = \frac{1}{c_+}\int \prod_{\sigma=1}^{n}\frac{\mathrm{d}x_\sigma\,\mathrm{d}\hat{x}_\sigma}{2\pi}\int\frac{\mathrm{d}y}{\sqrt{2\pi}}e^{-\frac{y^2}{2}}\Theta(y+b_0)$$
$$e^{-\frac{\beta}{2}\sum_\sigma[g(x_\sigma+b)-g(y+b_0)]^2}e^{-\frac{1}{2}(1-q)\sum_\sigma\hat{x}_\sigma^2+\frac{1}{2}(R^2-q)\sum_{\sigma,\sigma'}\hat{x}_\sigma\hat{x}_{\sigma'}+i\sum_\sigma\hat{x}_\sigma(x_\sigma-yR)} \tag{1.71}$$

In order to integrate out the $\hat{x}_\sigma$ we need to decouple the term $\hat{x}_\sigma\hat{x}_{\sigma'}$ through Hubbard-Stratonovich transform:

$$e^{-\frac{1}{2}(q-R^2)\sum_{\sigma,\sigma'}\hat{x}_\sigma\hat{x}_{\sigma'}} = \int Dt e^{(i\sqrt{q-R^2}\sum_\sigma\hat{x}_\sigma)t} \tag{1.72}$$

we recall that $Dt = \frac{\mathrm{d}t}{\sqrt{2\pi}}e^{-t^2/2}$.

Integrals over $\hat{x}_\sigma$ are now Gaussian and yield:

$$e^{-\mathcal{G}_r^+} = \frac{1}{c_+}\int Dy\int Dt\Theta(y+b_0)\prod_{\sigma=1}^{n}\int\frac{\mathrm{d}x_\sigma}{\sqrt{2\pi(1-q)}}e^{-\frac{(x_\sigma-yR+t\sqrt{q-R^2})^2}{2(1-q)}}e^{-\frac{\beta}{2}[g(x_\sigma+b)-g(y+b_0)]^2} \tag{1.73}$$

Performing the shift and re-scaling $x_\sigma \to x_\sigma\sqrt{1-q}+yR-t\sqrt{q-R^2}$ one gets:

$$e^{-\mathcal{G}_r^+} = \frac{1}{c_+}\int Dy\Theta(y+b_0)\int Dt\left[\int Dx e^{-\frac{\beta}{2}\left[g(x\sqrt{1-q}+yR-t\sqrt{q-R^2}+b)-g(y+b_0)\right]^2}\right]^n \tag{1.74}$$

Now we can compute the $n\to 0$ limit. We call $A = \int Dx e^{-\frac{\beta}{2}\left[g(x\sqrt{1-q}+yR-t\sqrt{q-R^2}+b)-g(y+b_0)\right]^2}$ and exploit the identity $A^n \sim 1+n\log A$ for $n\to 0$:

$$G_r^+ = \lim_{n\to 0}-\frac{1}{n}\log\left(1+n\frac{1}{c_+}\int Dy\Theta(y+b_0)\int Dt\log A\right) \tag{1.75}$$

Since n is small we can expand the first log around 1:

$$G_r^+ = -\frac{1}{c_+}\int Dy\Theta(y+b_0)\int Dt\log A \tag{1.76}$$

We recall our choice of the activation function $g(x) = \mathrm{sign}(x)$. The square-loss per sample in this case reads $\epsilon(\mathbf{w};\mathbf{S}) = 2\Theta(-(\mathbf{w}\cdot\mathbf{S}+b)(\mathbf{w}^0\cdot\mathbf{S}+b_0))$. We re-define it without the the factor 2 in order to count the number of mis-classified samples. G_r^+ becomes:

$$G_r^+ = -\frac{1}{c_+}\int_{-b_0}^{\infty}Dy\int_{-\infty}^{\infty}Dt\log\left(\int_{-\infty}^{\infty}Dx e^{-\beta\Theta\left(-(x\sqrt{1-q}+yR-t\sqrt{q-R^2}+b)(y+b_0)\right)}\right) \tag{1.77}$$

We define: $u = \frac{t\sqrt{q-R^2}-yR-b}{\sqrt{1-q}}$ and $H(x) = \int_x^\infty Dt = \frac{1}{2}\mathrm{erfc}\left(\frac{x}{\sqrt{2}}\right)$. The final form for $G_r^\pm$ reads:

$$G_r^+ = -\frac{1}{c_+}\int_{-b_0}^{\infty}Dy\int_{-\infty}^{\infty}Dt\log\left(e^{-\beta}+(1-e^{-\beta})H(u)\right) \tag{1.78}$$

$$G_r^- = -\frac{1}{c_-}\int_{-\infty}^{-b_0}Dy\int_{-\infty}^{\infty}Dt\log\left((e^{-\beta}-1)H(u)+1\right) \tag{1.79}$$

We now show the detailed computation for the entropic term G_0. Starting from (1.62) and substituting the RS ansatz one gets:

$$\mathcal{G}_0 = -\frac{1}{2}n(n-1)q\hat{q} - n\hat{R}R + \frac{1}{N}\log\int\prod_{\sigma=1}^{n}\mathrm{d}\mu(\mathbf{w}^\sigma)e^{\hat{q}\sum_{\sigma>\sigma'}\mathbf{w}^\sigma\mathbf{w}^{\sigma'}+\hat{R}\sum_\sigma\mathbf{w}^\sigma\mathbf{w}^0} \tag{1.80}$$

We decouple $\mathbf{w}^\sigma\mathbf{w}^{\sigma'}$ through Hubbard-Stratonovich:

$$e^{\hat{q}\sum_{\sigma>\sigma'}\mathbf{w}^\sigma\mathbf{w}^{\sigma'}} = e^{\frac{1}{2}\hat{q}\sum_{\sigma,\sigma'}\mathbf{w}^\sigma\mathbf{w}^{\sigma'}-\frac{1}{2}\hat{q}\sum_\sigma\mathbf{w}^\sigma\mathbf{w}^\sigma} = e^{-\frac{1}{2}\hat{q}\sum_\sigma\mathbf{w}^\sigma\mathbf{w}^\sigma}\int D\mathbf{z}e^{\sqrt{\hat{q}}\sum_\sigma\mathbf{w}^\sigma\mathbf{z}} \tag{1.81}$$

Then:

$$\mathcal{G}_0 = -\frac{1}{2}n(n-1)q\hat{q} - n\hat{R}R + \frac{1}{N}\log\int D\mathbf{z}\left(\int\mathrm{d}\mu(\mathbf{w})e^{\mathbf{w}(\hat{R}\mathbf{w}^0+\sqrt{\hat{q}}\mathbf{z}-\frac{1}{2}\hat{q}\mathbf{w})}\right)^n \tag{1.82}$$

Now we can take the $n\to 0$ limit:

$$G_0 = \lim_{n\to0}\frac{\mathcal{G}_0}{n} = -\hat{R}R + \frac{1}{2}q\hat{q} + \frac{1}{N}\int D\mathbf{z}\log\int\mathrm{d}\mu(\mathbf{w})e^{\mathbf{w}(\hat{R}\mathbf{w}^0+\sqrt{\hat{q}}\mathbf{z}-\frac{1}{2}\hat{q}\mathbf{w})}. \tag{1.83}$$

The last integral yields:

$$\int D\mathbf{z}\log\int\frac{\mathrm{d}\lambda}{4\pi i}e^{\frac{N\lambda}{2}}e^{-\frac{N}{2}\log[e(\lambda+\hat{q})]}e^{\frac{N}{2(\lambda+\hat{q})}(\hat{R}^2+\hat{q}\frac{\sum_i z_i^2}{N}+2\sqrt{\hat{q}}\hat{R}\frac{\sum_i\omega_i^0 z_i}{N})}. \tag{1.84}$$

Computing the integral over λ with a saddle point approximation we reduce the double integral to

$$N\left[\frac{\lambda}{2}-\frac{1}{2}\log[e(\lambda+\hat{q})]+\frac{\hat{R}^2}{2(\lambda+\hat{q})}+\frac{1}{2(\lambda+\hat{q})}\int D\mathbf{z}\left(\hat{q}\frac{\sum_i z_i^2}{N}+2\sqrt{\hat{q}}\hat{R}\frac{\sum_i\omega_i^0 z_i}{N}\right)\right], \tag{1.85}$$

where now λ denotes the saddle point value. The Gaussian integral over $\mathbf{z}$ is easily computed, and one gets as a result:

$$G_0 = -\hat{R}R + \frac{1}{2}q\hat{q} + \frac{\lambda}{2} - \frac{1}{2}\log(\lambda+\hat{q}) + \frac{1}{2}\frac{\hat{R}^2+\hat{q}}{(\lambda+\hat{q})} - \frac{1}{2}. \tag{1.86}$$

Saddle Point Equations We look for a stationary point of the variational free energy to fix the value of the order parameters at equilibrium. We then set to 0 the derivatives of the variational free energy with respect to the order parameters $(q,\hat{q},R,\hat{R},b)$ and the additional Lagrange multiplier λ that we introduced to enforce the spherical constraint for the students'

weights.

$$R = \hat{R}(1-q) \tag{1.87}$$

$$q = (\hat{q} + \hat{R}^2)(1-q)^2 \tag{1.88}$$

$$\hat{R} = \alpha \frac{e^{-\frac{b_0^2 q}{2(q-R^2)}}}{2\pi\sqrt{1-q}} \left\{ \frac{\rho}{c_+} \int_{-\infty}^{\infty} Dt \frac{e^{-v^2/2 + \frac{b_0 R}{\sqrt{q-R^2}} t}}{(e^{\beta}-1)^{-1} + H(v)} - \frac{1-\rho}{c_-} \int_{-\infty}^{\infty} Dt \frac{e^{-v^2/2 + \frac{b_0 R}{\sqrt{q-R^2}} t}}{(e^{-\beta}-1)^{-1} + H(v)} \right\} \tag{1.89}$$

$$\hat{q} = \frac{\alpha}{2\pi(1-q)} \left\{ \frac{\rho}{c_+} \int_{-b_0}^{\infty} Dy \int_{-\infty}^{\infty} Dt \frac{e^{-u^2}}{[(e^{\beta}-1)^{-1} + H(u)]^2} + \right.$$
$$\left. + \frac{1-\rho}{c_-} \int_{-\infty}^{-b_0} Dy \int_{-\infty}^{\infty} Dt \frac{e^{-u^2}}{[(e^{-\beta}-1)^{-1} + H(u)]^2} \right\} \tag{1.90}$$

$$0 = \frac{\rho}{c_+} \int_{-b_0}^{\infty} Dy \int_{-\infty}^{\infty} Dt \frac{e^{-u^2/2}}{(e^{\beta}-1)^{-1} + H(u)} + \frac{1-\rho}{c_-} \int_{-\infty}^{-b_0} Dy \int_{-\infty}^{\infty} Dt \frac{e^{-u^2/2}}{(e^{-\beta}-1)^{-1} + H(u)} \tag{1.91}$$

For any given choice of the hyperparameters $b_0, \rho_{\text{train}}, T$ and α (in SM these are called control parameters), we can solve these selfconsistent equations numerically, to obtain the order parameters, with particular interest in (R, b). These order parameters contain all the information about the typical behavior of the learned model at the end of the training. In the following we show how to extract them.

Train and Generalization metrics

Train and generalization metrics are all derived from the order parameters at equilibrium. The average **train error** per sample can be evaluated as:

$$\epsilon_t = \frac{1}{P} \langle\!\langle \mathbb{E}_T[\mathcal{E}(\mathbf{w}, b)] \rangle\!\rangle_{\mu_{\text{train}}} \tag{1.92}$$

where $\mathbb{E}_T[\dots]$ denotes the average over the realizations of the thermal noise. Explicitly:

$$\epsilon_t = \frac{1}{N\alpha} \langle\!\langle \frac{1}{Z} \int \mathrm{d}\mu(\mathbf{w}) \int \mathrm{d}\mu(b) \mathcal{E}(\mathbf{w}, b) e^{-\beta \mathcal{E}(\mathbf{w}, b)} \rangle\!\rangle_{\mu_{\text{train}}} = -\frac{1}{N\alpha} \langle\!\langle \frac{\partial}{\partial \beta} \log Z \rangle\!\rangle = \frac{1}{N\alpha} \frac{\partial(\beta F)}{\partial \beta} \tag{1.93}$$

$$\epsilon_t = -\frac{1}{\alpha} \frac{\partial}{\partial \beta} \left\{ G_0(q, \hat{q}, R, \hat{R}) - \alpha \rho_{\text{train}} G_r^+(q, R, b) - \alpha(1-\rho_{\text{train}}) G_r^-(q, R, b) \right\} \tag{1.94}$$

evaluated in the saddle point. The volume term G_0 doesn't depend on the temperature and we get:

$$\epsilon_t = \rho_{\text{train}} \frac{\partial G_r^+(q, R, b)}{\partial \beta} + (1-\rho_{\text{train}}) \frac{\partial G_r^-(q, R, b)}{\partial \beta} \tag{1.95}$$

$$= \frac{\rho_{\text{train}}}{c_+} \int_{-b_0}^{\infty} Dy \int_{-\infty}^{\infty} Dt \frac{1 - H(u)}{1 + (e^{\beta}-1) H(u)} + \frac{1-\rho_{\text{train}}}{c_-} \int_{-\infty}^{-b_0} Dy \int_{-\infty}^{\infty} Dt \frac{H(u)}{e^{\beta} + (1-e^{\beta}) H(u)} \tag{1.96}$$

For any metric $M(\mathbf{w}, b; \mathbf{S})$, its generalization value is:

$$M_g = \langle\langle \mathbb{E}_T[M(\mathbf{w}, b; \mathbf{S})] \rangle\rangle_{\mu_{\text{train}}} \rangle_{\mu_{\text{test}}} \tag{1.97}$$

The idea behind the computation of generalization metrics is to add one sample that was not observed during training and evaluate the performance of the trained student on it. In practice, we can define the **generalization error** as:

$$\epsilon_g = \langle\langle \mathbb{E}_T[\epsilon(\mathbf{w}, b; \mathbf{S})] \rangle\rangle_{\mu_{\text{train}}} \rangle_{\mu_{\text{test}}} \tag{1.98}$$

The first average is on the train-set and it yields the saddle-point equations shown in the previous section. The second average, on the test-set is needed to evaluate the trained student on the new, unseen sample. Explicitly:

$$\epsilon_g = \langle\langle \frac{1}{Z} \int \mathrm{d}\mu(\mathbf{w}) \int \mathrm{d}\mu(b) \epsilon(\mathbf{w}, b; \mathbf{S}) e^{-\beta \mathcal{E}(\mathbf{w}, b)} \rangle\rangle_{\mu_{\text{train}}} \rangle_{\mu_{\text{test}}} \tag{1.99}$$

$$= \lim_{n \to 0} \langle\langle Z^{n-1} \int \mathrm{d}\mu(\mathbf{w}) \int \mathrm{d}\mu(b) \epsilon(\mathbf{w}, b; \mathbf{S}) e^{-\beta \sum_{\ell=1}^{N\alpha} \epsilon(\mathbf{w}, b; \mathbf{S}^\ell)} \rangle\rangle_{\mu_{\text{train}}} \rangle_{\mu_{\text{test}}} \tag{1.100}$$

$$= \lim_{n \to 0} \int \mathrm{d}\mu_{\text{test}}(\mathbf{S}) \int \mathrm{d}\mu_{\text{bias}}(\{\mathbf{S}^\ell\}) \int \prod_{\sigma=1}^{n} \mathrm{d}\mu(b_\sigma)\, \mathrm{d}\mu(\mathbf{w}^\sigma) \epsilon(\mathbf{w}^1, b_1; \mathbf{S}) e^{-\beta \sum_\ell \sum_\sigma \epsilon(\mathbf{w}^\sigma, b_\sigma; \mathbf{S}^\ell)} \tag{1.101}$$

$$= \lim_{n \to 0} \int \mathrm{d}\mu_{\text{test}}(\mathbf{S}) \int \prod_{\sigma=1}^{n} \mathrm{d}\mu(b_\sigma)\, \mathrm{d}\mu(\mathbf{w}^\sigma) \epsilon(\mathbf{w}^1, b_1; \mathbf{S}) e^{-N\alpha\rho \mathcal{G}_r^+(\{\mathbf{w}^\sigma, b_\sigma\}) - N\alpha(1-\rho)\mathcal{G}_r^-(\{\mathbf{w}^\sigma, b_\sigma\})} \tag{1.102}$$

One can evaluate:

$$\int \mathrm{d}\mu_{\text{test}}(\mathbf{S}) \epsilon(\mathbf{w}^1, b_1; \mathbf{S}) = \frac{\rho_{\text{test}}}{c_+} \mathrm{I}_+ + \frac{1 - \rho_{\text{test}}}{c_-} \mathrm{I}_- \tag{1.103}$$

Where we have defined:

$$\mathrm{I}_\pm = \int D\mathbf{S} \Theta\left(\pm \frac{\mathbf{w}^0 \cdot \mathbf{S}}{\sqrt{N}} \pm b_0 \right) \epsilon(\mathbf{w}^1, b_1; \mathbf{S}) \tag{1.104}$$

In the following we show the computation for I_+, the one for I_- follows the same lines.

$$\mathrm{I}_+ = \int D\mathbf{S} \Theta\left(\frac{\mathbf{w}^0 \cdot \mathbf{S}}{\sqrt{N}} + b_0 \right) \epsilon(\mathbf{w}^1, b_1; \mathbf{S}) \tag{1.105}$$

$$= \int \frac{\mathrm{d}x\, \mathrm{d}\hat{x}}{2\pi} \int \frac{\mathrm{d}y\, \mathrm{d}\hat{y}}{2\pi} e^{ix\hat{x} + iy\hat{y}} \frac{1}{2} [g(x + b_1) - g(y + b_0)]^2 \Theta(y + b) \int D\mathbf{S} e^{-\frac{i}{\sqrt{N}} (\mathbf{w}^1 \hat{x} + \mathbf{w}^0 \hat{y}) \cdot \mathbf{S}} \tag{1.106}$$

Integrating over $\mathbf{S}, \hat{x}, \hat{y}$ one gets:

$$\mathrm{I}_+ = \int Dx \int_{-b_0}^{\infty} Dy \frac{1}{2} [g(x\sqrt{1 - R_1^2} + yR_1 + b_1) - g(y + b_0)]^2 = \tag{1.107}$$

$$= \int Dx \int_{-b_0}^{\infty} Dy \Theta(-(x\sqrt{1 - R_1^2} + yR_1 + b_1)(y + b_0)) = \tag{1.108}$$

$$= \int_{-b_0}^{\infty} Dy \int_{-\infty}^{u'} Dx = \int_{-b_0}^{\infty} Dy (1 - H(u')) \tag{1.109}$$

with $u' = \frac{-yR_1 - b_1}{\sqrt{1-R_1^2}}$

Computing also I_- we get:

$$\epsilon(R_1, b_1) = \frac{\rho_{\text{test}}}{\frac{1}{2}\text{erfc}\left(\frac{-b_0}{\sqrt{2}}\right)} \int_{-b_0}^{\infty} Dy \int_{-\infty}^{u'} Dx + \frac{1-\rho_{\text{test}}}{1-\frac{1}{2}\text{erfc}\left(\frac{-b_0}{\sqrt{2}}\right)} \int_{-\infty}^{-b_0} Dy \int_{u'}^{\infty} Dx \tag{1.110}$$

Thus we can rewrite the generalization error:

$$\epsilon_g = \lim_{n\to 0} \int \prod_{\sigma} \mathrm{d}b_\sigma \int \prod_{\sigma>\sigma'} \frac{\mathrm{d}Q_{\sigma,\sigma'}\,\mathrm{d}\hat{Q}_{\sigma,\sigma'}}{2\pi i} \int \prod_{\sigma} \frac{\mathrm{d}R_\sigma\,\mathrm{d}\hat{R}_\sigma}{2\pi i} \times \tag{1.111}$$

$$\times\, \epsilon(R_1, b_1) e^{-N\alpha\rho\mathcal{G}_r^+(\{Q_{\sigma,\sigma'},R_\sigma,b_\sigma\}) - N\alpha(1-\rho)\mathcal{G}_r^-(\{Q_{\sigma,\sigma'},R_\sigma,b_\sigma\}) + N\mathcal{G}_0(\{Q_{\sigma,\sigma'},\hat{Q}_{\sigma,\sigma'},R_\sigma,\hat{R}_\sigma\})} \tag{1.112}$$

Following the same lines of the Replica Calculation performed in the previous section one gets:

$$\epsilon_g = \epsilon(R, b) \tag{1.113}$$

with (R, b) parameters at equilibrium i.e. the ones that solve the saddle-point equations.

As introduced in Sec.1.2, all the generalization metrics that we investigate in the manuscript can be expressed in terms of **True Positive Rate** (Recall, r) and the **True Negative Rate** (Specificity, s). Here we show the derivation for these two metrics:

$$r = \frac{\left\langle\left\langle\left\langle \mathbb{E}_T\left[\left[1-\Theta\left(-\left(\frac{\mathbf{w}\cdot\mathbf{S}}{\sqrt{N}}+b\right)\left(\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}+b_0\right)\right)\right]\Theta\left(\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}+b_0\right)\right]\right\rangle\right\rangle_{\mu_{\text{train}}}\right\rangle_{\mu_{\text{test}}}}{\rho_{test}} \tag{1.114}$$

$$= \frac{1}{c_+}\int D\mathbf{S}\Theta\left(\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}+b_0\right)\left[1-\Theta\left(-\left(\frac{\mathbf{w}\cdot\mathbf{S}}{\sqrt{N}}+b\right)\left(\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}+b_0\right)\right)\right]\Big|_{S.P.} \tag{1.115}$$

$$= 1 - \frac{1}{c_+}\int D\mathbf{S}\Theta\left(\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}+b_0\right)\Theta\left(-\left(\frac{\mathbf{w}\cdot\mathbf{S}}{\sqrt{N}}+b\right)\left(\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}+b_0\right)\right)\Big|_{S.P.} \tag{1.116}$$

$$= 1 - \frac{1}{c_+}\mathrm{I}_+(R, b, b_0)\Big|_{S.P.} \tag{1.117}$$

The derivation follows the same lines of the one for the Generalization Error. We stress that metric is evaluated at the saddle point for the order parameters.

$$s = \frac{\left\langle\left\langle\left\langle \mathbb{E}_T\left[\left[1-\Theta\left(-\left(\frac{\mathbf{w}\cdot\mathbf{S}}{\sqrt{N}}+b\right)\left(\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}+b_0\right)\right)\right]\Theta\left(-\frac{\mathbf{w}^0\cdot\mathbf{S}}{\sqrt{N}}-b_0\right)\right]\right\rangle\right\rangle_{\mu_{\text{train}}}\right\rangle_{\mu_{\text{test}}}}{1-\rho_{test}} \tag{1.118}$$

$$= 1 - \frac{1}{c_-}\mathrm{I}_-(R, b, b_0)\Big|_{S.P.} \tag{1.119}$$

1.3.3 Theoretical Results

Here we present the theoretical results of our work on Anomaly-Detection Class Imbalance in Exactly Solvable Models. The main contributions of our work are the following :

- By solving the teacher-student (TS) spherical perceptron in the presence of AD Imbalance, we provide an **interpretable framework to characterize AD Imbalance**. This allows us to elucidate the role of three sources of imbalance: the intrinsic imbalance ρ_0, the training imbalance, ρ_{train}, and the test set imbalance, ρ_{test}. As a function of these quantities, we examine how various commonly used performance metrics are able to track both the overlap of the Student with the Teacher model, and how the Student bias reproduces that of the Teacher. The bias of the Student is more sensitive to ρ_{train} than to the bias of the Teacher.
- Should one attempt at re-balancing the training set to have $\rho_{\text{train}} = 0.5$, as it is usually done? Challenging the common intuition that a perfectly balanced training set is optimal, we find that **the optimal value of ρ_{train} is not 0.5**. Factors influencing this value and its relevance include the abundance of data, the amount of noise in the dynamics, and the bias of the teacher.
- **Dynamics with lower noise are less susceptible to CI**. We identify two distinct regions. For low noises, the performance is optimal, and largely unaffected by the noise level. For large noises, the performance degrades and becomes sensitive to additional amounts of noise. This effect correlates with the degree of imbalance in the system, and is related to how well the student can reconstruct the teacher bias.

When the teacher bias is known

We begin by considering a simpler scenario where the student's bias is not learned but fixed at $b = b_0$, corresponding to the situation in which the student has prior knowledge of the teacher's bias. This case is particularly insightful because it reveals some underlying symmetries of the problem and clarifies the concept of *informative samples* introduced in Sec. 1.3.1.

Optimal training. In this setup, the self-consistent equations presented in Sec. 1.3.2 simplify, reducing to just the first four equations, since the student's bias b is fixed and does not need to be fixed self-consistently. The relevant information is thus captured entirely by the teacher-student overlap R. A key consequence of this fixed bias is that **translations of the student's decision boundary are not permitted**. Therefore, the alignment between teacher and student depends solely on the density of samples near the teacher boundary, regardless of their class labels. When $b_0 \neq 0$, one class becomes more informative about the teacher's direction, meaning that samples closer to the teacher's boundary provide more information about its position. This becomes particularly evident here: when $b_0 \neq 0$, one class is inherently more informative, and to maximize the overlap R, **the optimal training set would ideally consist solely of samples from the minority class**. This phenomenon is illustrated in Fig. 1.3–(left), where the resulting R from training is plotted against ρ_{train} for various values of the teacher's bias b_0. We observe that as $|b_0|$ increases, this effect becomes more pronounced, as the minority class samples become increasingly concentrated near the teacher's boundary.

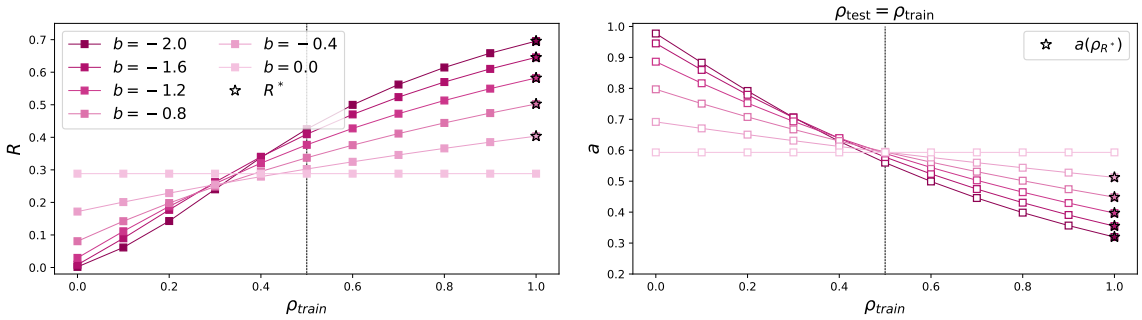


Figure 1.3: **Overlap and accuracy on the spherical teacher-student perceptron, with the constraint $b = b_0$.** *(left)* Teacher-student overlap R as a function of ρ_{train}, for $\alpha = 0.7$ and $T = 0.5$. Stars indicate the point where the overlap is maximized. The vertical line indicates $\rho_{\text{train}} = 0.5$. *(right)* Test-set accuracy a with $\rho_{\text{test}} = \rho_{\text{train}}$ as a function of ρ_{train}, for $\alpha = 0.7$ and $T = 0.5$. Stars indicate the point where the overlap is maximized which correspond to a low accuracy on the test set.

Invariance under sample reflection. The case $b = b_0 = 0$ is particularly interesting because it highlights a symmetry in the problem. Here, both classes contribute equally to informativeness, meaning there is no advantage in having more samples from one class over the other. This symmetry is reflected in the flat curve in Fig. 1.3–(left). It also manifests in the free energy function, where the two energetic terms $\mathcal{G}_r^{\pm}$, defined in Eq. (1.51), become equal when $b_\sigma = b_0 = 0$. In fact, by applying the change of variables $\mathbf{S} \to -\mathbf{S}$ in the integral, we recover $\mathcal{G}_r^-$ from $\mathcal{G}_r^+$ and vice versa. This symmetry arises because the Boltzmann weight of each sample is identical regardless of its label, meaning that as long as the total number of samples remains fixed, the free energy remains the same. In essence, when $b = 0$, there exists a bijection between flipping the labels and flipping the samples $\mathbf{S}$. Thus, in the second integral of Eq. (1.50), imposing a label flip also imposes a reflection in the data space, leading to the problem's invariance under sample reflection.

Test accuracy. Another important insight from this simplified case is that evaluating simple accuracy a on a test set with the same imbalance as the training set ($\rho_{\text{test}} = \rho_{\text{train}}$) can be misleading. We observe that the value of ρ_{train} which maximizes accuracy often corresponds to a lower overlap R. This discrepancy arises because a higher density of samples near the boundary increases the likelihood of misclassification, lowering accuracy even when the alignment between teacher and student is quite strong. This effect is demonstrated in Fig. 1.3–(right), where the accuracy on the test set is plotted against ρ_{train}.

Learning the bias

The most intriguing phenomenology arises when the full problem is considered, *i.e.*, when the student is also required to learn the bias. In the following, we summarize the key observations and insights from this scenario.

Good and bad models, Energy-Entropy interplay. We begin by investigating the influence of ρ_{train} on the learned model. Figure 1.5 (a)-(b), reports the solution of the self-consistent equations for the overlap R and the learned bias b as a function of ρ_{train}, for multiple choices of

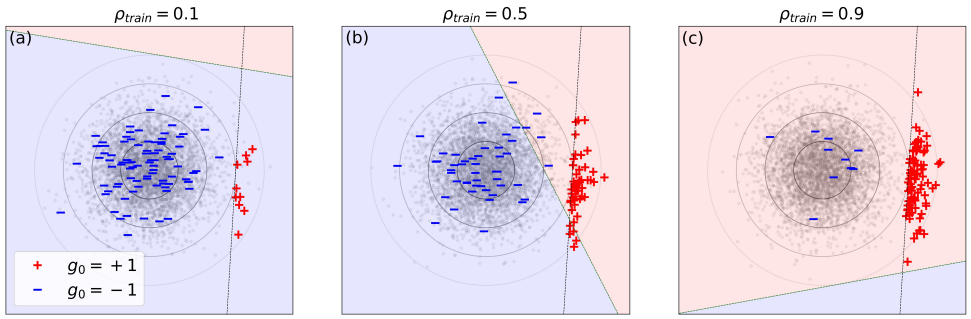


Figure 1.4: **Geometrical interpretation of learning an Anomaly Detection task under class imbalance**, with fixed ρ_0, and $\rho_{\text{train}} = 0.1, 0.5, 0.9$. Normal examples (negative label, $g_0^\ell = -1$) are represented with blue – symbols, anomalies (positive label, $g_0^\ell = +1$) with red +. Shaded grey points depict the underlying Gaussian data distribution and grey circles locate contours at $1\sigma, 2\sigma$ and 3σ (σ is the standard deviation). The black dashed line represents the teacher decision boundary which determines the ground-truth labels and the colored regions of the plane depict the classes predicted by the student (the model being trained). The three examples contain the same number of misclassified examples, but the learned model is very different. When the training set is strongly imbalanced ((a) and (c)) the student has an entropic incentive to learn a strong bias, completely discarding the alignment with the teacher: learning only a bias that matches the train imbalance is statistically favored due to the large number of possible directions for the student's decision plane that achieve a low error. Learning on a balanced training set (b) forces the student to learn the direction of the teacher because of the higher cost of mis-classifying one of the two classes.

the teacher bias b_0. Learning under strong imbalance leads to a model that has a strong bias and low alignment with the teacher, this is a *bad* model since it is not able to reproduce b_0 and $\mathbf{w}_0$ correctly. Meanwhile, learning on a more balanced training set, leads to a *good* model that is able to better reproduce the teacher's labeling rule, learning a good overlap R and not being overly biased. This phenomenon can be geometrically interpreted (Fig. 1.4a,c): since the loss counts the number of misclassified examples, a dummy model that has a strong bias and always predicts the majority class pays a small price in terms of loss. However, it is statistically favored with respect to a model with $b = b_0$, since with $|b| \gg |b_0|$, there is a very large number of weight configurations $\mathbf{w}$ that allow for the same small training error, while with $b = b_0$ the number of weight configurations giving a small error is much lower. This is an example of what is called energy-entropy interplay [CABJ20]: solutions with large b have an entropic advantage (there are more of them), at the expense of a few misclassifications (what is sometimes called an *energetic cost*). As ρ_{train} becomes more balanced, the energetic cost increases, eventually overcoming the entropic advantage. We also note that, while the situations in Figs. 1.4a,c are similar with what regards the training errors, they are of course very dissimilar when one looks into the generalization error (accuracy curve in Fig. 1.5d).
Finally, we highlight that training at $\rho_{\text{train}} = \rho_0$ is always sub-optimal, both in terms of R and of b.

Performance metrics, which one? When testing a trained model, it is common practice to build a balanced test set with samples not observed during training, *i.e.*, $\rho_{\text{test}} = 0.5$. Another practice is to leave the distribution untouched, $\rho_{\text{test}} = \rho_0(b_0)$. How are different performance

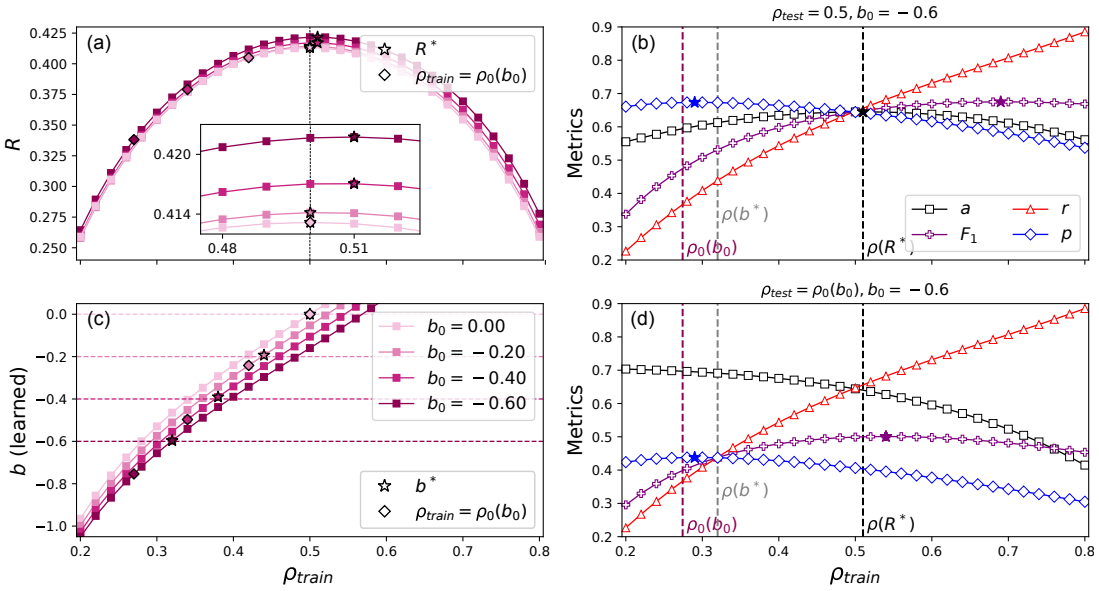


Figure 1.5: **Analytical results as a function of ρ_{train}**, for $\alpha = 1.1$ and $T = 0.5$. *(a)* Student overlap R, for $b_0 = 0, -0.2, -0.4, -0.6$ ($\rho_0 = 0.5, 0.42, 0.34, 0.27$). Stars indicate the point where the overlap is maximized, diamonds indicate the performance at $\rho_{\text{train}} = \rho_0$. The vertical line indicates $\rho_{\text{train}} = 0.5$. The inset is a zoom. *(b)* Same as (a), but for the student bias b. The horizontal lines indicate b_0. Now the stars indicate the points where $b = b_0$, and the diamonds indicate the b that would be learned if one trained with $\rho_{\text{train}} = \rho_0$. *(c)* Accuracy, recall, precision and F1 score, for $b_0 = -0.6$, $\rho_{\text{test}} = 0.5$. The stars indicate the peak of each curve. The vertical lines indicate ρ_0, the imbalance $\rho(b^*)$ at which the bias is optimal, and that at which the overlap is optimal, $\rho(R^*)$. *(d)* Same as (c), but for $\rho_{\text{test}} = \rho_0(b_0)$.

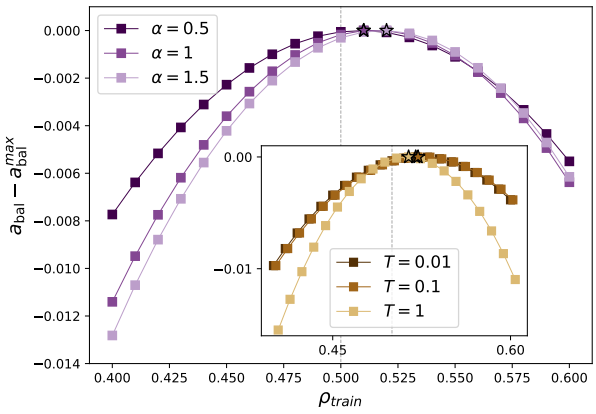


Figure 1.6: **Dependence on ρ_{train} for different α or T.** The optimal balanced accuracy (a_{bal}). We plot a_{bal} as a function of ρ_{train}, shifted so that all the curves peak at 0. The vertical dotted lines indicate $\rho_{train} = 0.5$. *(Main)* study at $b_0 = -1$ and $T = 0.5$. Varying α changes the position of the peak, as well as how fast the performance decreases when leaving the peak. *(Inset)* same b_0 and fixed $\alpha = 1.1$, varying T. The cases $T = 0.01$ and $T = 0.1$ are almost impossible to distinguish because they both correspond to the low-temperature region (Fig. 1.8). For high T the curvature is larger.

metrics affected by the test imbalance, and which metric is best able to identify a *good* model in terms of R and b? Figure 1.5(c,d) report different performance metrics for models trained with varying ρ_{train} and tested with $\rho_{test} = 0.5$ and $\rho_{test} = \rho_0$.
The recall is trivially maximized for $\rho_{train} = 1$, since this generates a dummy model which identifies anything as an anomaly. The sensitivity s has the opposite trend (not shown), being trivially maximized for $\rho_{train} = 0$.
The balanced accuracy (not explicitly plotted, since it coincides with a when $\rho_{test} = 0.5$) is the quantity that best reproduces $R(\rho_{train})$, and shares with $R(\rho_{train})$ the feature of not depending on ρ_{test}.
The accuracy for $\rho_{test} = \rho_0$ is maximized at small ρ_{train}, because this generates a model which always guesses the majority class. This can also be seen from the expression of the accuracy in terms of recall and specificity *i.e.* $a = \rho_{test} r + (1 - \rho_{test}) s$, where if $\rho_{test} < 0.5$ the contribution of the s dominates.
The trend of the precision seems independent of ρ_{test}, though its specific value is. p peaks between ρ_0 and $\rho(b^*)$, making it the best candidate to identify b_0.
Finally, F_1, when calculated with $\rho_{test} = 0.5$, peaks at a value representing a low R and a bias with sign opposite to b_0. It is instead more informative when $\rho_{test} = \rho_0$ is used, since it peaks at a value close to $\rho(R^*)$.
In summary, we identify a_{bal} as the most suitable metric to identify the overlap, due to the qualitative agreement with R and the independence from ρ_{test}. The metric which best identifies the optimal b is instead the precision.

Optimal train imbalance. We already noted that training at $\rho_{train} = \rho_0$ never gives the best model (in terms of best R nor b). We now turn to $\rho_{train} = 0.5$, which is commonly believed to lead to optimal generalization performances, and the most common choice in CI reweighting/resampling schemes. We challenge this assumption, showing that the optimal train-

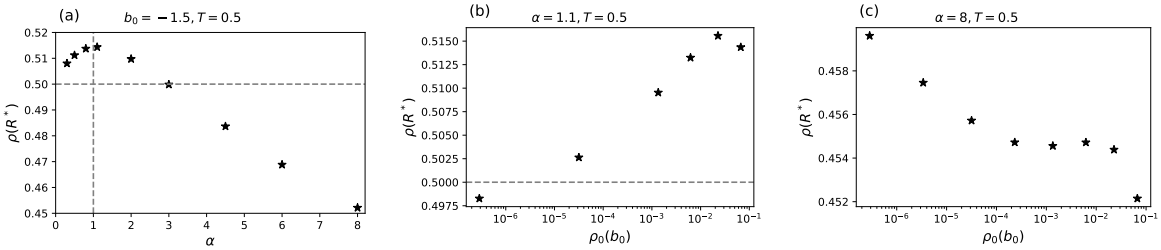


Figure 1.7: **Optimal ρ_{train} as function of the control parameters α and $\rho_0(b_0)$.** *(a)* $\rho(R^*)$ as function of data abundance α. *(b)* $\rho(R^*)$ as function of the intrinsic imbalance ρ_0 (controlled by b_0), for $\alpha = 1.1$. *(c)* Same as (b), for $\alpha = 8$. The dependencies are non monotonous, indicating a highly non-trivial behavior. Dashed grey lines highlight the values $\rho_{\text{train}} = 0.5$ or $\alpha = 1$.

ing imbalance, $\rho_{\text{train}}(R^*) = \text{argmax}_{\rho_{\text{train}}}(R)$, is different from 0.5. This is true for the overlap (Fig. 1.5a-inset), and it is also true for the best proxy of the overlap, a_{bal}. Fig. 1.6 shows that, when $b_0 < 0$, a_{bal} on the test set peaks at $\rho_{\text{train}} > 0.5$, *i.e.* when there are slightly more of the anomalous examples. This is consistent with previous empirical observations on SVMs and Random Forests, which found that $\rho_{\text{train}} = 0.5$ is not the optimal training ratio [KAE22].

We also look into the influence of the degree of under-parameterization (α) and the amount of noise in the dynamics (T) on the value of $\rho_{\text{train}}(R^*)$: for the values considered in Fig. 1.6, increasing α and decreasing T make the curves more tilted, shifting the optimal train imbalance, and the curves more peaked, thus increasing the penalty for choosing $\rho_{\text{train}} \neq \rho_{\text{train}}(R^*)$.

In Fig. 1.7 we further investigate this effect, showing that $\rho_{\text{train}}(R^*)$ depends on α, T and b_0; that the effect is non-monotonic; and it can shift $\rho_{\text{train}}(R^*)$ both to values > 0.5 (as in Fig. 1.6) and < 0.5.
We argue that this is the result of two competing effects. On one side, as depicted in Fig. 1.1d, minority class examples are more informative, so it is more convenient to train with more of those (*i.e.* increase ρ_{train}). On the other side, if $|b_0|$ is large enough, there is a large region $\mathcal{R}$ between typical negatives and (informative) positives that is empty of points, thus allowing for many possible hyperplane directions $\mathbf{w}$ that separate the training set. This means that a large fraction of student models with $|b| < |b_0|$ will result in a small error. Since there are many more weight configurations allowing for $|b| < |b_0|$ than weight configurations allowing $b = b_0$, configurations with a wrong b and $\mathbf{w}$ are entropically favored. One way to decrease this entropic contribution is to fill the region $\mathcal{R}$ with negatives, *i.e.* increase the proportion of negatives (decrease ρ_{train}).

Interplay between noise and CI. We investigate the impact of the amount of noise in the dynamics (T) on the quality of the learned model. Fig. 1.8 shows a crossover around a temperature T^*. Below T^* the performance is optimal, and largely unaffected by the noise level. Above T^*, the performance degrades and becomes sensitive to any additional amount of noise. By comparing with Fig. 1.6-inset, we see that the performance in the high-noise regime is connected to a lower tolerance to non-optimal values of ρ_{train}.

A similar effect has already been observed in the teacher-student perceptron, from a dynamical

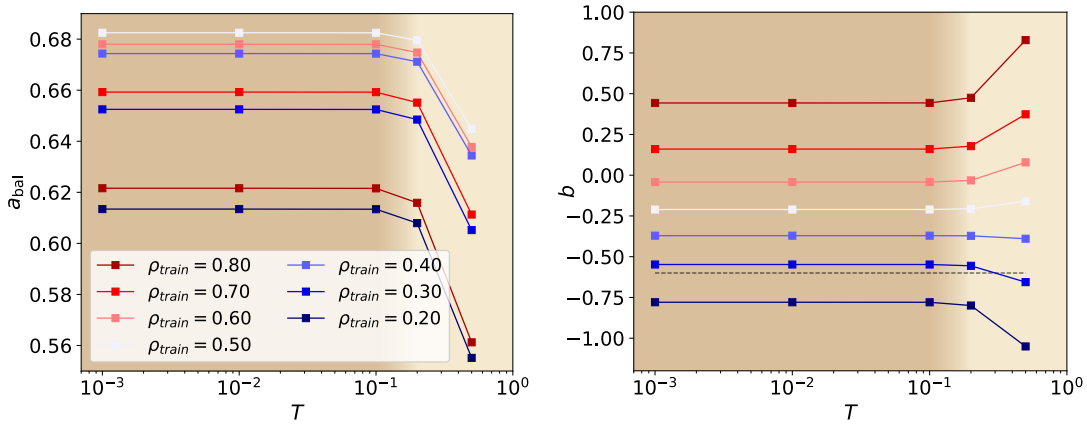


Figure 1.8: **Performance as function of T.** *(Left)* Balanced accuracy as a function of temperature T, for $\alpha = 1.1$. The teacher bias is $b_0 = -0.6$ (dotted horizontal line in the inset, $\rho_0 = 0.27$). *(Right)* Same, for the learned bias b.

perspective, by studying the regimes of Stochastic Gradient Descent (SGD) as shown in [SW24]. They identify a Gradient Descent-like regime with low noise and optimal performances and a noise-dominated regime where performance deteriorates as noise increases. Here, we elucidate the interplay between the noise level and the training imbalance. In particular, we observe that ρ_{train} determines the evolution of the student's bias with T: increasing T favors the entropic contribution discussed in Fig. 1.4 and, depending on the value of ρ_{train}, this results in a more or less biased model. This can be again interpreted in terms of an energy-entropy interplay: noise in the dynamics places greater importance on the statistical abundance of solutions rather than minimal error, thus favoring overly-biased dummy models. This effect is more pronounced when considering strong imbalance scenarios, *i.e.* ρ_{train} approaching 1 or 0.

1.3.4 Experiments

We consider two experimental setups to assess: (i) the influence of learning dynamics and the read-out (activation) function on our results within a controlled scenario that mirrors theoretical computations, and (ii) the effects of dataset characteristics and model choice in a scenario more akin to practical machine learning applications.

Perceptron TS: In this setup, we use a Spherical Perceptron within a Teacher-Student framework. The model is linear, with the weights' norm constrained to be $O(1)$, and employs a sigmoid activation function. The model is trained through the minimization of the L2 loss. This configuration closely mirrors the theoretical setup, with two key distinctions: the use of a sigmoid activation function that outputs continuous values within the $(0, 1)$ range and SGD learning dynamics, similar to real machine learning practices. It is important to note that the choice of a continuous activation function is essential to enable gradient descent dynamics, as a discontinuous function, such as a sign function, would result in gradients that are either zero or infinite. To produce binary labels, teacher outputs are discretized, assigning a label of 1 for values above thr = 0.5 and 0 otherwise. The SGD dynamic shares some characteristics with the Langevin dynamics used in the theoretical derivation, as both implement gradient descent on

the training loss with added stochastic noise. The key difference lies in the correlated nature of the SGD noise, which arises from mini-batch gradient estimates.

To approximate theoretical conditions as closely as possible, we set the data dimension to $N = 5000$. We observe that, by further increasing N, our results remain stable, suggesting that this setting is close enough to the infinite-dimensional limit. We train the student model by performing multiple passes on the whole training set (epochs) until the training loss has converged. This corresponds to the "end of training" (equilibrium) regime assumed in theoretical computations. The noise level in SGD is governed by the learning rate and batch size, which can be approximately related to the temperature introduced in the main text as $T \sim \mathrm{lr}/\mathrm{BS}$ [JKA+17].

For each combination of control parameters $(b_0, \rho_{\text{train}}, T, \alpha)$, we perform multiple runs, resampling both the dataset and teacher weights to compute the quenched average over the data distribution. Results on a balanced test set ($\rho_{\text{test}} = 0.5$) are illustrated in Fig. 1.9-(left). We observe trends that are qualitatively compatible with theoretical results, and most importantly, we find a non-trivial maximum of the metrics at $\rho^* \neq 0.5$. In experiments, we also evaluate the AUC metric since a threshold is needed to discretize the output of the learned perceptron. We observe that it is rather insensitive to imbalance, confirming the findings of [LPCM24]. Figure 1.10 reports the trends of balanced accuracy and student's bias versus effective temperature. The experimental trends align qualitatively with the theoretical ones, identifying the low-noise and high-noise regions separated by T^* as introduced in Sec. 1.3.2.

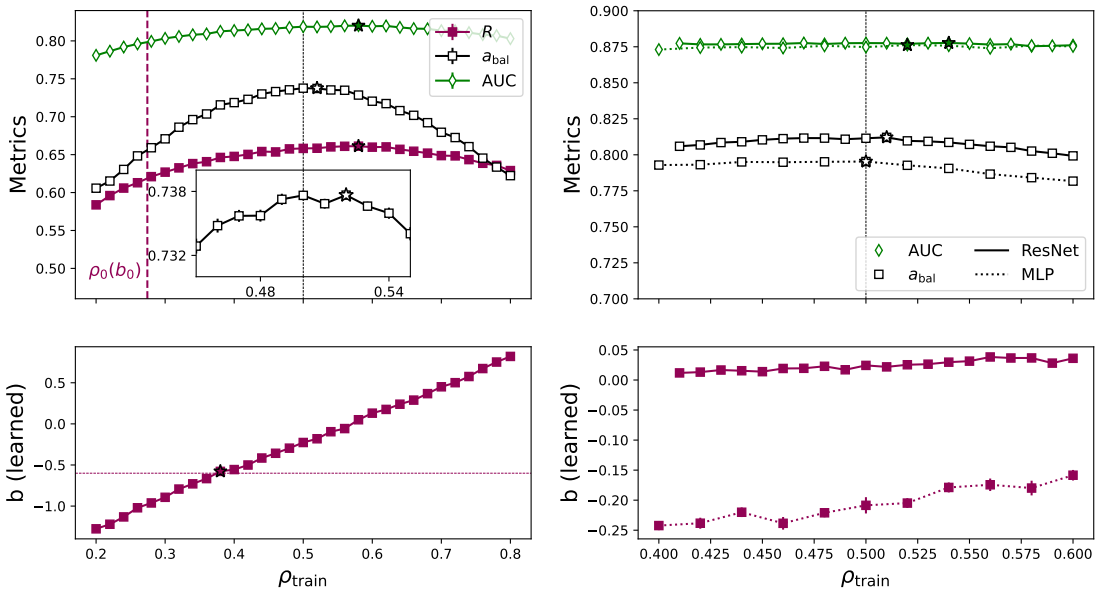


Figure 1.9: (*left*) **Perceptron TS**. $b_0 = -0.6$, $\alpha = 2.0$. Effective temperature $T = \frac{\mathrm{lr}}{\mathrm{BS}} = \frac{0.5}{20} = 2.5 \cdot 10^{-2}$. Each point represents the average over 40 re-samplings of the data and the error-bar represents its relative standard error. (*right*) **MLP and ResNet on AD CIFAR-10**. SGD optimizer, with momentum = 0.02 and weight decay 0.01. Each point represents the average over 10 re-samplings of the data and the error-bar represents its relative standard error.

MLP and ResNet on AD CIFAR-10: in this setup, we employ a real-world anomaly detection dataset: Anomaly Detection CIFAR-10, a standard benchmark for anomaly detection tasks (see

e.g. [RCBH20]). This dataset involves re-labeling the original CIFAR-10 classes such that the first class (Airplanes) is designated as the anomaly (label +1), while all other classes are labeled as normal samples (label 0). CIFAR-10 consists of 60,000 samples, with 6,000 samples per class, structured and non-independent by design. Defining one class as the anomaly sets the intrinsic imbalance to $\rho_0 = 0.1$. To explore various values of ρ_{train}, we perform sub-sampling on the dataset according to the desired level of imbalance, keeping the total number of training samples fixed at $N_{\text{train}} = 6000$.

We evaluate two representative models: an MLP with one hidden layer of 16 neurons, and a pre-trained ResNet34 [HZRS15] in which only the final linear layer is trained on the anomaly detection task, while all other layers remain frozen. Training is conducted via L2 loss minimization, using SGD for learning dynamics.

In this experimental setup, not all theoretical hyper-parameters can be controlled. For instance, ρ_0 is fixed by the dataset, and defining b_0 is not feasible. Additionally, tuning α is challenging for both theoretical and practical reasons. Theoretically, the definition of α differs significantly from that in our analytical model, as the data dimensionality and model parameter count are no longer in one-to-one correspondence. Practically, varying N_{train} or adjusting the MLP's hidden layer size is constrained by limited data availability and the risk of overfitting. Nonetheless, for each parameter configuration, we re-train the models multiple times, re-sampling data from the original CIFAR-10.

Results on a balanced test set, shown in Fig. 1.9-(right), reveal a phenomenology qualitatively consistent with theoretical predictions. However, the effect strength and available statistics limit the conclusiveness of these findings.

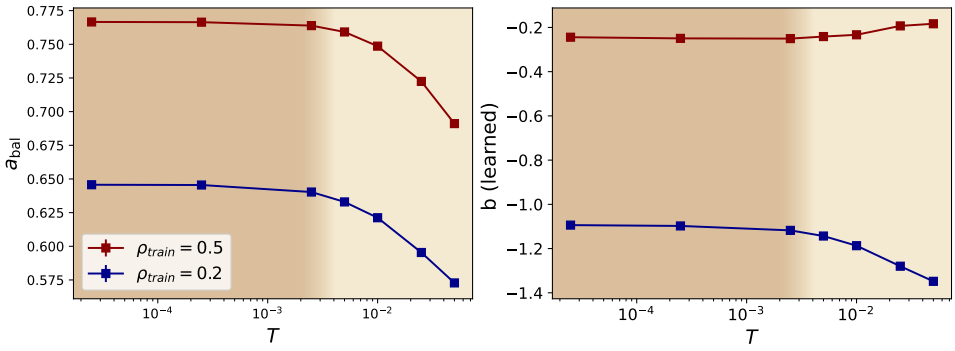


Figure 1.10: **Perceptron TS vs T.** Temperature is varied in SGD experiments by tweaking the the mini-batch size. The learning rate is fixed to be lr = 0.05 and the mini-batch size varies $BS = \{2000, 200, 20, 10, 5, 2, 1\}$.

1.4 Conclusion

In this chapter, we analyzed the effect of AD Imbalance on learning, through exact analytical calculations grounded in statistical physics methods. We constructed an interpretable framework that clarifies the influence of various factors on the learning outcome.

In addition to the train and test imbalance, ρ_{train} and ρ_{test}, **in AD Imbalance we identified an intrinsic imbalance, $\boldsymbol{\rho_0}$, over which practitioners have no control**. If data generation is unbiased, no re-balancing of the class distributions is performed and the dataset faithfully represents the deployment distribution, then one has $\rho_0 = \rho_{\text{train}} = \rho_{\text{test}}$.

Varying ρ_{train} corresponds to re-balancing the training distribution. Since our results are in the asymptotic data limit, they equally represent the effect of both class reweighting and resampling. Note, however, that these two re-balancing strategies influence SGD differently [FBJL23].

We have showed that **the value of $\boldsymbol{\rho_{\text{train}}}$ which maximizes the overlap between teacher and student is generally not 0.5**. This is consistent with previous empirical work [KAE22], where on different kind of architectures it was shown that re-sampling using some $\rho_{\text{train}} < 0.5$ was consistently optimal over a broad range of tasks and models. The case $\rho_{\text{train}} > 0.5$ was however not explored. A trend was observed: as data is initially more abundant (corresponding to larger α for us), more re-sampling can be done. Our results show that the picture can in general be more complex. In fact, this **deviation from $\boldsymbol{\rho^*_{\text{train}} = 0.5}$ depends non-linearly on $\boldsymbol{\rho_0}$ and on $\boldsymbol{\alpha}$**. While α indicates how much data is available in comparison with the model size, in our linear classifier it also indicates the dimensionality of the input space. Therefore, we cannot disentangle whether this effect is due to model size or to input dimensionality. We also found that **the importance of this deviation is amplified in dynamics with a strong noise** (*e.g.* large learning rate), with small-noise dynamics leading to better solutions than larger-noise ones, with a clear separation between two regimes.

This asymmetry is at least in part a consequence of the fact that, in AD Imbalance, examples from different classes are intrinsically not equally informative. While this asymmetry was, to our knowledge, not observed in previous work on MG Imbalance, we believe that similar deviations from $\rho_{\text{train}}(R^*) = 0.5$ can also be observed, in cases where different classes inform differently on the classification boundary (*e.g.* a class having smaller variance). In particular, in MG classification, we conjecture that this asymmetry could also be observed in the absence of imbalance. In fact, while in AD Imbalance, $\rho_{\text{train}}(R^*) \neq 0.5$ is a feature of the imbalance, in MG classification it can be a feature of the data structure.

From the point of view of what happens to the training landscape when varying ρ_{train}, we observed that it causes smooth variations in the solutions, with no abrupt changes even when values such as $\rho_{\text{train}} = \rho_0$ or 0.5 are crossed. We noticed such an **absence of phase transitions** also when tuning ρ_0 and ρ_{test}.

Varying ρ_{train} and ρ_{test}, and the evaluation metrics, informs us on what each metric reproduces. **The balanced accuracy seems the best proxy for the teacher overlap $\boldsymbol{R}$**, while the quantity that best reflects the bias is the precision p.

Finally, we outline four key directions for further research:

- **Data Distribution:** the first step to enhance connection with realistic settings is to assess the validity of the AD Imbalance assumption. This involves determining to what extent the overlap between distributions applies in real datasets. Future research could focus on modeling more complex data distributions that incorporate meaningful structures, moving beyond the current assumption of independent data points.

- **Loss Function:** in the current model, the loss function does not consider the distance between misclassified samples and the student boundary, and margins are absent. Exploring the use of a hinge loss could provide new insights. The introduction of margins may significantly alter the learning landscape, leading to a stronger alignment with the teacher model and potentially different qualitative behavior.
- **Improved Models:** extending the analysis to deeper and more sophisticated architectures, such as kernel machines or fully connected networks, would improve the connection to practical scenarios. While deep networks remain analytically challenging, certain limits, like the infinite width scenario, are well-understood. For instance, networks with infinitely wide hidden layers have been linked to kernel machines [BCP20].
- **Multiclass Classification:** extending our results to multi-class scenarios would allow the exploration of various forms of class imbalance. This includes investigating different label distributions across multiple classes, which could provide deeper insights into the impact of imbalance on learning.

Chapter 2

Rotation-equivariant networks for glassy liquids

Contents

In this chapter, we explore the emergence of amorphous order in supercooled liquids using roto-translational equivariant graph neural networks. We begin by introducing the phenomenology

of supercooled liquids, discussing the main limitations of current theoretical approaches and numerical simulations, along with key open questions. Next, we review research that leverages machine learning models to study the structure of these materials and to correlate structural features with dynamical properties. We then turn to equivariant neural networks, providing the mathematical background on group theory and the theoretical framework needed to understand the construction of such models. Finally, we present our work on SE(3)-equivariant GNNs for glassy liquids, achieving state-of-the-art results in predicting particle mobilities from static structures. Our findings reveal valuable insights into the role of directional information, temperature generalization, and transferability.

2.1 Supercooled Liquids

Typically, when a liquid is cooled, it reaches a characteristic temperature T_m, known as the melting temperature, at which it undergoes a phase transition and crystallizes. However, for a large number of liquids, a carefully controlled cooling process can allow them to enter a metastable phase known as a *supercooled liquid*. This phase is amorphous, meaning that **simple structural descriptors do not reveal long-range order**, unlike in a crystal, where a periodic structure is formed. To achieve this state, crystallization must be avoided by preventing the nucleation of the stable crystal phase from the metastable liquid phase. This is typically done by cooling the liquid quickly enough to prevent the formation of crystal nuclei, but not so rapidly as to drive the system out of equilibrium. Although supercooled liquids are often described as being in equilibrium, they are technically out of equilibrium, as the true thermodynamic equilibrium below the melting temperature is the crystalline state, while the liquid remains metastable. However, it is possible to experimentally stabilize a supercooled liquid in such a way that time-translation invariance (and thus the fluctuation-dissipation theorem) holds. In this case, no experimental observation can easily distinguish that the system is in a metastable state rather than in true equilibrium.

The supercooled liquid phase is a precursor to the *dynamic glass transition*. As the supercooled liquid is further cooled, it eventually reaches a point, commonly referred to as T_g, where it can no longer equilibrate, moving out of equilibrium. At this stage, on any experimentally accessible timescale, the liquid behaves as an amorphous solid: it no longer flows, despite showing no obvious structural order. Though obtaining a supercooled liquid requires careful cooling, glassy materials are ubiquitous in daily life, with examples including window glasses (silica glass), glassy gels [Daw02], protein-systems [FML$^+$02], and metallic glasses [Joh02]. For many materials, achieving a crystalline state is even more challenging than forming a glass: even slow cooling in laboratory conditions may result in a glassy solid as T_g is approached.

Supercooled liquids and the glass transition have been studied by physicists for a long time, yet despite their many real-world applications, a comprehensive theoretical framework that fully describes these phenomena is still lacking. In the following, we will delve into the phenomenology of supercooled liquids to highlight the significance of studying them. For an in-depth overview of phenomenological perspectives and theoretical approaches to supercooled liquids, see the excellent reviews by Cavagna [Cav09] and Arceri [ALBB20], which inspired the following section.

2.1.1 Phenomenology

Viscosity and relaxation time, Angell plot In supercooled liquids, as the temperature decreases, the molecular motion slows down, and the liquid begins to resemble a solid in terms of its mechanical rigidity. This dramatic slowing down of molecular motion is reflected in both the viscosity and the relaxation time. The *viscosity* (η) measures the resistance of a liquid to flow. It quantifies the internal friction that arises when adjacent layers of fluid move relative to each other. It can be defined as the ratio of the applied shear stress σ_{xy} and the rate of shear strain $\dot{\gamma}$:

$$\eta = \frac{\sigma_{xy}}{\dot{\gamma}} \tag{2.1}$$

The *relaxation time* (τ_α) refers to the timescale over which a system relaxes the stress imposed by a shear transformation. In the context of supercooled liquids, it is often associated with the time required for the molecular structure to de-correlate from its initial configuration. The two quantities can be related in a simple Maxwell model for a liquid:

$$\eta = G_\infty \tau_\alpha \tag{2.2}$$

where G_∞ is the elastic modulus representing the material's ability to resist deformation in the short timescale before relaxation occurs.

As the temperature decreases in a supercooled liquid, both viscosity and relaxation time increase dramatically. This means that as the liquid cools, it becomes increasingly difficult for the molecules to rearrange, and the liquid behaves more like an elastic solid over long timescales. This increase is commonly illustrated for many glass-forming liquids using the Angell plot (2.1), which displays the logarithm of viscosity or relaxation time as a function of inverse temperature, normalized by the dynamical glass transition temperature T_g. This temperature is conventionally defined as the point where the viscosity reaches 10^{13}Poise, corresponding to when the material's relaxation time exceeds the longest measurable experimental timescale.

$$\tau_\alpha(T < T_g) > t_{exp} \tag{2.3}$$

The increase is so dramatic that viscosity can change by 14 orders of magnitude with just a "small" variation in temperature around T_g. This indicates a significant qualitative change in the material's state.

The Angell plot clearly distinguishes between two distinct families of materials:

Strong liquids: the relationship between temperature and relaxation time (or viscosity) follows an Arrhenius law:

$$\tau_\alpha \sim \exp\left(\frac{\Delta}{k_B T}\right) \tag{2.4}$$

Here, Δ is an activation energy barrier, and the liquid's molecular rearrangements occur via thermally activated processes that remain constant as temperature decreases. In these liquids, viscosity and relaxation time increase linearly with the inverse of the temperature.

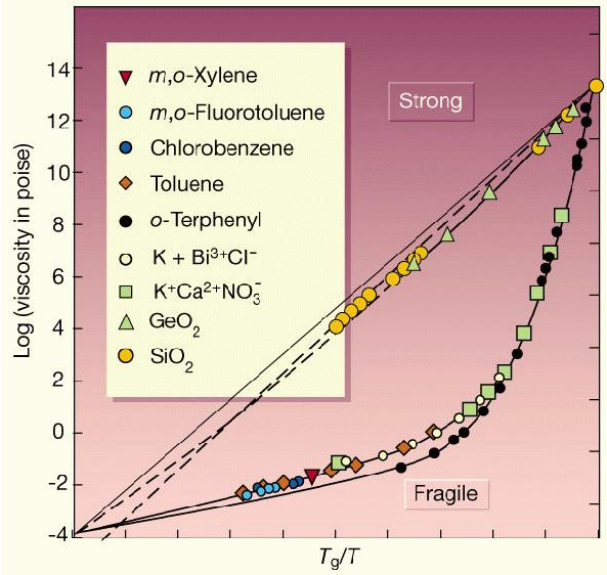


Figure 2.1: **Angell's plot**: Strong liquids exhibit approximate linearity (Arrhenius behaviour), indicative of a temperature-independent activation energy. Fragile liquids exhibit super-Arrhenius behaviour, their effective activation energy increasing as temperature decreases. (Reproduced from [DS01])

Fragile liquids: the increase in viscosity and relaxation time is much more dramatic and follows a super-Arrhenius behavior. This is commonly captured by the Vogel-Fulcher-Tammann (VFT) law:

$$\tau_\alpha = \tau_0 \exp\left(\frac{A}{T - T_0}\right) \tag{2.5}$$

The VFT law suggests a divergence at a finite temperature T_0, although this cannot be observed experimentally. More importantly, the VFT law implies that energy barriers increase as the temperature decreases. This rise in energy barriers suggests that glass formation in fragile supercooled liquids is a collective phenomenon, where molecular rearrangements require cooperative movement of increasingly larger groups of molecules. For this reason, **fragile glass formers** are particularly intriguing, and they **will be the focus of our study**.

Excess Entropy, Kauzmann Temperature Further experimental evidence supporting the idea of **cooperative rearranging regions in fragile liquids** comes from their thermodynamic behavior. As the temperature decreases, the entropy of the supercooled liquid can be compared to that of the crystalline state by defining the excess entropy as:

$$\Delta S(T) = S_{LQ}(T) - S_{CR}(T) \tag{2.6}$$

The entropy of the supercooled liquid decreases more rapidly than that of the crystal, making the excess entropy a decreasing function of temperature (see fig. 2.2). This quantity can be calculated only down to the glass transition temperature T_g, as the system falls out of equilibrium below this point. However, by extrapolating the excess entropy to lower temperatures, it is found that for some materials, the excess entropy vanishes at a finite temperature T_K, referred to as the

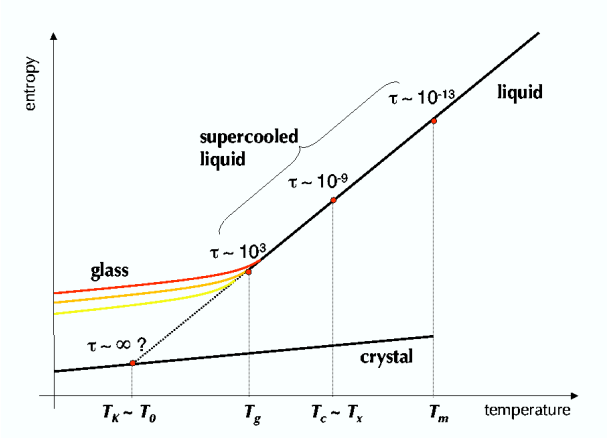


Figure 2.2: **Excess entropy**: Entropy of the liquid and the corresponding crystal as a function of temperature. As temperature decreases, the entropy of the supercooled liquid decreases more rapidly than that of the crystal. This trend can be observed down to T_g, below which the liquid falls out of equilibrium, illustrated by the colored curves representing the out-of-equilibrium glass. The extrapolated entropy curve of the liquid intersects the crystal entropy at T_k. (Reproduced from [Cav09])

Kauzmann temperature. This temperature is very close to T_0, the point at which the relaxation time diverges according to the VFT law for many glass formers [Ang97, RA98].

This coincidence becomes even more intriguing when considering that the excess entropy is thought to provide a good approximation of the configurational entropy in supercooled liquids. A widely accepted model in the community for describing deeply supercooled liquids is the Goldstein scenario, where, near T_g, the supercooled liquid navigates phase space through activated jumps between different amorphous minima of the free energy landscape. These minima are separated by sufficiently high energy barriers to associate each state with a distinct minimum. This suggests that the entropy of the liquid can be divided into two components: the vibrational entropy (S_{vib}), which is associated with the motions within a single minimum and is roughly equivalent to the entropy of the crystal (S_{CR}), and the configurational entropy (S_c), which quantifies the number of amorphous states that the liquid can explore in an ergodic manner. Therefore, the excess entropy can be approximated as:

$$\Delta S(T) = S_{LQ}(T) - S_{CR}(T) \sim S_c(T) \tag{2.7}$$

The vanishing of excess entropy at T_K implies that the configurational entropy also vanishes, meaning the number of amorphous states that the liquid can explore below T_K becomes sub-exponential. This suggests a breakdown of ergodicity and points to a true thermodynamic phase transition. Unfortunately, since empirical measurements cannot be performed below T_g, this transition cannot be directly observed. Nevertheless, the fact that $T_K \sim T_0$ links thermodynamic considerations to purely dynamical behavior, **suggesting a connection between the slowdown of dynamics**, the reduction in configurational entropy, **and the breakdown of ergodicity**.

A breakdown of ergodicity, along with an infinite relaxation time, implies infinite energy barriers. In other words, the system should become trapped in one of the lowest-lying minima below T_K.

But why would the energy barriers between amorphous minima become infinite at T_K? The physical mechanism driving this drastic phenomenon likely lies in the cooperative motion of particles in real space. **The emergence of a static correlation length**, which increases (and ideally diverges) as temperature decreases, **would bridge these two observations and suggest the formation of an amorphous order**. Such a scenario would classify the glass transition as a thermodynamic phase transition.

Static Correlation Function When examining standard two-point correlation functions, no clear evidence of a growing length scale is observed. The simplest way to structurally characterize a homogeneous and isotropic liquid is through the radial distribution function $g(r)$, which gives the probability of finding a particle at a distance r from a reference particle:

$$g(r) = \frac{1}{N}\frac{1}{4\pi r^2\rho}\langle\sum_i^N\sum_{j>i}^N\delta(r-r_{ij})\rangle \tag{2.8}$$

where N is the total number of particles, ρ the density and $r_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$. The radial distribution function is an effective tool for distinguishing different phases: the more structured the system, the more pronounced and well-defined the peaks in $g(r)$. In liquids, one typically observes a series of smooth peaks with decreasing height, corresponding to successive coordination shells, reflecting the absence of long-range order due to the disordered nature of the liquid. In contrast, in crystals, the peaks are sharp and persistent over long distances, indicating the presence of long-range order characteristic of crystalline structures.

From the experimental point of view the more accessible quantity to measure is the static structure factor $S(q)$. It provides the same structural information as $g(r)$ but in momentum space, and is related to $g(r)$ by the following Fourier integral:

$$S(q) = 1 + 4\pi\rho\int_0^\infty \mathrm{d}r r^2\frac{\sin qr}{qr}(g(r)-1) \tag{2.9}$$

The evolution of $S(q)$ with temperature is reported in Fig. 2.3. No significant changes are observed, and notably, **there is no evidence of a growing length scale** that can be extracted from this data.

Dynamical Correlation Functions and MSD When examining dynamical observables, a clear indication of dynamical slowdown as the transition is approached becomes evident. More significantly, this slowdown is not only quantitative but also qualitative, reflecting a change in the underlying mechanism of particle rearrangement. This shift can be detected by studying dynamic correlation functions, particularly the *incoherent intermediate scattering function*:

$$F_s(\mathbf{q},t) = \frac{1}{N}\sum_i^N\langle\delta\rho_i(\mathbf{q},t)\delta\rho_i(-\mathbf{q},0)\rangle \tag{2.10}$$

where $\delta\rho_i(\mathbf{q},t) = \exp\left[-i\mathbf{q}\cdot\mathbf{r}_i(t)\right]$ represents the $\mathbf{q}$-component of the Fourier transform of the fluctuations of density associated with particle i.

This function measures how quickly the system decorrelates from its initial configuration. By fixing $\mathbf{q}$, we select a corresponding length scale in real space, effectively probing particles that

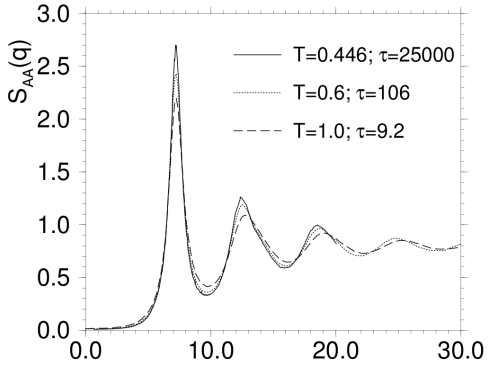


Figure 2.3: **Static Structure Factor**: measured in the numerical simulation of a Lennard-Jones liquid at three different temperatures. Minor changes are observed in the structure while the relaxation time changes by almost 4 orders of magnitude. (Reproduced from [Kob03])

have moved a distance comparable to this scale. If $\|\mathbf{r}_i(t) - \mathbf{r}_i(0)\|$ is larger than $1/\|\mathbf{q}\|$ then $\delta\rho_i(\mathbf{q},t)\delta\rho_i(\mathbf{q},0) \sim 0$. Thus, this function provides insight into the fraction of particles that have moved a given distance after time t. The behavior of such function as the temperature approaches the glass transition is shown in Fig. 2.4-(left).

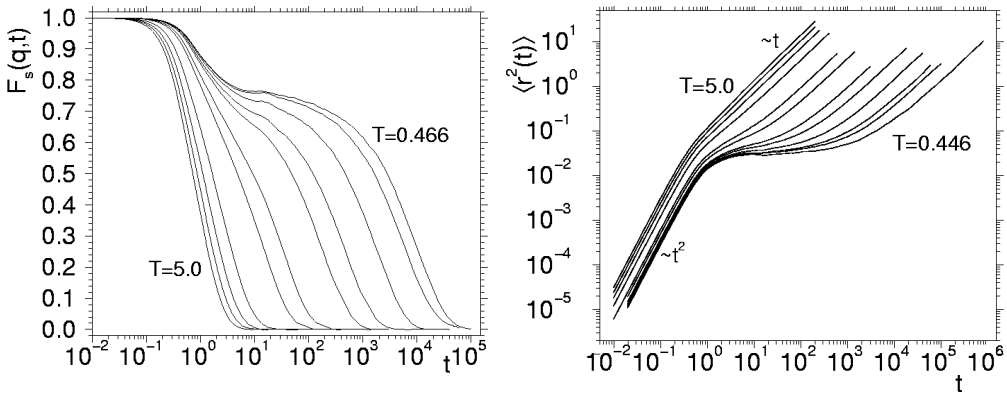


Figure 2.4: *(Left)* **Incoherent Intermediate Scattering Function**: At high temperatures, relaxation is exponential, characterized by a single timescale. As temperature approaches T_g, a two-step relaxation process emerges, indicating two distinct timescales. *(Right)* **Mean Squared Displacement**: Illustrates similar behavior as seen in $F_s(q,t)$; at high temperatures, normal diffusion is observed, while at lower temperatures, a plateau forms, highlighting the two-step relaxation and introducing the concept of the "cage" effect. (Reproduced from [KA95a] and [KA95b])

At high temperatures the system behaves like a typical liquid. In this regime, particles are able to move freely, and the decay of the incoherent intermediate scattering function is simple and fast. $F_s(\mathbf{q},t)$ shows a single-exponential decay, which reflects the fact that particles decorrelate from their initial positions on relatively short timescales. The motion of particles is primarily

diffusive, and relaxation is uniform across the system.

As the liquid is cooled and approaches T_g, the dynamics begin to change significantly, and $F_s(\mathbf{q}, t)$ exhibits a two-step decay: at short timescales, $F_s(\mathbf{q}, t)$ decays quickly due to local vibrational motion of particles within the "cages" formed by their neighbors. This initial decay is often referred to as β-relaxation and reflects the rapid, short-range motion of particles that remain trapped by their surrounding neighbors. After this fast initial decay, it reaches a plateau, indicating that the system is temporarily stuck in a configuration where particles are caged by their neighbors. During this time, particles do not move far enough to escape their cages, and the system remains highly correlated over intermediate timescales. At longer timescales, the system undergoes a second, much slower decay, known as α-relaxation. This corresponds to the cooperative motion of particles, allowing them to escape their cages and rearrange the structure of the liquid. The time required for this relaxation increases dramatically as temperature decreases, reflecting the overall slowdown of dynamics near the glass transition.

This two-step decay of $F_s(\mathbf{q}, t)$ is a qualitative fingerprint of glass transition as it relates the increase of relaxation time to a new mechanism determining the particles re-arrangement.

The interpretation of the two-step decay is closely tied to the so-called cage picture, which is also evident when examining the mean squared displacement (MSD) of a particle. For short times, the MSD exhibits ballistic motion ($\sim t^2$), followed by a plateau, reflecting that the particle is trapped within a local cage of neighboring particles. This plateau reflects the confinement of the particle, with very limited movement. Eventually, at longer times, the particle escapes this cage, and the MSD resumes a diffusive behavior ($\sim t$). Interestingly, although the cage picture is useful for describing real-space systems, the two-step relaxation phenomenon also appears in mean-field models [CHS93], where real-space structures and cages do not physically exist. This suggests that, while the cage effect provides an intuitive explanation for particle dynamics in supercooled liquids, the underlying mechanism behind the two-step relaxation is likely more fundamental and extends beyond real-space interactions.

2.1.2 Dynamical Heterogeneities

A relevant feature of the incoherent intermediate scattering function at low temperatures is that the tail of the α-relaxation does not follow a simple exponential decay, as one might expect. Instead, it takes the form of a stretched exponential, $\exp[-(t/\tau_\alpha)^\beta]$ with $\beta < 1$ [GSTC96]. Interestingly, the exponent β decreases as the temperature is lowered, indicating a growing deviation from standard exponential relaxation the deeper the system enters the supercooled phase. This behavior suggests an **increasingly heterogeneous dynamics in space, with different regions of the material relaxing on different timescales**. The observed curve would then represents an average of these relaxation processes occurring across various parts of the system at different rates.

A similar conclusion can be drawn from the behavior of the self-part of the van Hove function, defined as:

$$G_s(\mathbf{r}, t) = \left\langle \frac{1}{N} \sum_i \delta(\mathbf{r} - [\mathbf{r}_i(t) - \mathbf{r}_i(0)]) \right\rangle \tag{2.11}$$

This function measures the distribution of particle displacements over a given timescale. At high temperatures, $G_s(\mathbf{r}, t)$ closely follows a Gaussian distribution, characteristic of a standard diffusive process. However, as the temperature approaches T_g, the tails of the distribution become significantly broader, deviating from the Gaussian shape. These fat tails are better described by an exponential law, indicating the presence of a population of particles that move considerably farther than the rest, marking them as distinctly more mobile.

Another phenomenon supporting the previous explanations is the decoupling between the self diffusion coefficient (D) and viscosity. In typical fluids, these two quantities are related by the Stokes-Einstein relation, expressed as:

$$D \sim \frac{T}{\eta} \tag{2.12}$$

This relation assumes that molecular diffusion and viscosity are governed by the same relaxation processes, so as the liquid's viscosity increases, its diffusivity decreases proportionally. However, in supercooled liquids, this proportionality breaks down as the temperature decreases towards the glass transition. Specifically, the viscosity increases much more rapidly than the diffusion coefficient reduces. In other words, while the liquid becomes extremely viscous and resists flow, some particles remain unexpectedly mobile, moving through the system more easily than the relation would predict.

All these observations point to the existence of heterogeneous dynamics in supercooled liquids, a feature referred to as *dynamic heterogeneity*. This phenomenon has been extensively observed in experiments on glass-forming liquids. As the glass transition is approached, the dynamics become more spatially heterogeneous, with regions of the material displaying vastly different relaxation behaviors [CWCK$^+$10, BBB$^+$11, CLB$^+$21, ALBB20].

In order to characterize these dynamic heterogeneities, four-point correlation functions were introduced [DGP99, DFGP02]. These functions enable the extraction of a typical length scale and time scale associated with dynamical domains and allow the observation of a growing length scale which is inherently dynamical. Choosing as observable the density fluctuation one defines:

$$G_4(r,t) = \langle \delta\rho(0,0)\delta\rho(0,t)\delta\rho(r,0)\delta\rho(r,t)\rangle - \langle \delta\rho(0,0)\delta\rho(0,t)\rangle\langle \delta\rho(r,0)\delta\rho(r,t)\rangle \tag{2.13}$$

to unveil cooperative dynamics these functions examine the behavior at two spatial locations, separated by r, at two different instants of time, separated by t. It trivially corresponds to inspecting the correlations of particles displacements over a time-scale t at two positions in space with distance r. In this perspective one can define a field $\varphi(x,t) = \delta\rho(x,0)\delta\rho(x,t)$ and note that its average is just a dynamical correlation function (independent on space because the system is statistically homogeneous):

$$C(t) = \langle \varphi(x,t)\rangle \tag{2.14}$$

On the other hand, G_4 measures the *spatial fluctuations* of the same field, namely:

$$G_4(r,t) = \langle \varphi(0,t)\varphi(r,t)\rangle - \langle \varphi(0,t)\rangle\langle \varphi(r,t)\rangle = \langle \varphi(0,t)\varphi(r,t)\rangle - C(t)^2 \tag{2.15}$$

It is now clear that this function quantifies the spatial fluctuations of dynamical observables around the mean value which is represented by $C(t)$. This is precisely what we were interested

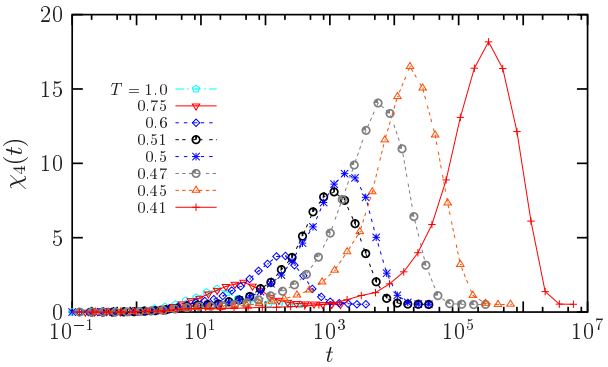


Figure 2.5: **Fluctuations of the Self-overlap function**: for each temperature, $\chi_4(t)$ has a maximum, which shifts to larger times and has a larger value when T is decreased, revealing the increasing length-scale of dynamic heterogeneity in supercooled liquids approaching the glass transition. (Reproduced from [ALBB20])

in quantifying. By integrating in space the four-point correlation function one obtains the corresponding susceptibility:

$$\chi_4(t) = \int \mathrm{d}r G_4(r,t) \tag{2.16}$$

This quantity allows to extract the typical volume of correlated regions and the time-scale over which correlation, and thus cooperative dynamics is maximal. By looking at Fig. 2.5 one can notice that the maximum of $\chi_4(t)$ increases when lowering the temperature and it is always located at $t = \tau_\alpha$. This reveals that the largest cooperative regions are observed during the α-relaxation and that their typical size is increasing with $1/T$ meaning that there's an inherently **dynamical lenght-scale ξ_D which grows approaching the glass transition**.

2.1.3 Numerical simulations

Numerical simulations are an essential tool for studying supercooled liquids, as they provide direct access to microscopic dynamical observables. In these simulations, the trajectories of individual particles can be tracked, allowing for the detailed characterization of their dynamics over timescales spanning multiple α-relaxation periods. Such resolution is challenging to achieve experimentally, where timescales of tens of seconds often preclude microscopic observations. Although the temperatures accessible in simulations are typically higher than their experimental counterparts, recent advancements in numerical techniques have enabled the simulation of sufficiently low temperatures, allowing systems to enter the deep-supercooled regime where clear signatures of the glass transition emerge [BR22].

Classical numerical simulations of supercooled liquids primarily use molecular dynamics (MD) approaches. In MD, once the interaction potential between particles is specified, the time evolution is computed typically within either the Canonical (NVT) ensemble, which employs a thermal bath to keep temperature constant, or the Micro-Canonical (NVE) ensemble, which follows a deterministic, energy-conserving discretization of Newton's equations.

More advanced simulation techniques can also resort to Monte-Carlo methods with moves carefully designed in order to avoid trapping in meta-stable states and slowing down of the simula-

tions [JMG+98]. One example is the Swap Monte Carlo method [GP01, BCNO16, NBC17] which allows to generate equilibrated configurations of supercooled liquids even below T_g. Unlike standard simulations, the swap Monte Carlo algorithm introduces unphysical moves by randomly exchanging the identities of particle pairs. When constructed to satisfy detailed balance, these swap moves allow the system to reach thermal equilibrium, significantly expanding the range of accessible temperatures compared to conventional simulation methods [SGB22].

Several studies have examined the impact of different microscopic dynamics on the behavior of glass-formers. Interestingly, while differences in dynamical fluctuations were observed, these dynamics were found to yield equivalent results at the level of averaged dynamical behavior. See *e.g.* [BBB+07].

Regarding models used for the simulations, we can distinguish three categories following the classification of [BR22]. The first category consists of realistic off-lattice models, which simulate the microscopic details of interactions in common glass-forming substances. These models typically involve complex features, such as non-spherical particle shapes, rotational degrees of freedom, and long-ranged interactions [vBKvS90]. As a result, simulating the glass-forming behavior of these systems is computationally intensive, with both molecular dynamics and Monte Carlo simulations becoming significantly more time-consuming compared to simpler models. The second category includes in silico glass formers, where spherical particles interact via short-range potentials [KA95a]. These models often incorporate some degree of polydispersity[1] and carefully tuned interactions to avoid crystallization. Despite their relative simplicity, in silico models exhibit nearly the full range of non-trivial behaviors observed in more complex molecular glass-formers, making them highly efficient for numerical simulations while still retaining the key features necessary to study glassy dynamics. The third category comprises fully coarse-grained lattice models [BM01], where particles are placed on a lattice and evolve through local Monte Carlo moves without any inter-particle forces. These models are valuable due to their simplicity, which allows for the direct application of analytical techniques, making them useful for theoretical investigations of glass formation. In our work, **we employ numerical simulations of** a glass-forming liquid from the second category, specifically the **3D Kob-Andersen Lennard-Jones mixture** [KA95a], a widely used model system for studying glassy dynamics.

Inherent Structures Numerical simulations provide a variety of techniques to filter out thermal fluctuations from particles' dynamics and extract information about the underlying potential energy landscape. One such technique is the identification of inherent structures [SW82]. This involves an instantaneous quench, where the system's temperature is set to zero, effectively eliminating particle velocities. The particle positions are then gradually adjusted to relax the forces and allow the system to converge to a local minimum of the potential energy, typically close to the initial configuration [BKG+06]. In the Goldstein picture introduced in section 2.1.1 a glass-forming liquid in the deeply supercooled phase spends most of the time near a minimum of the potential energy landscape, with thermal fluctuations causing particles to vibrate around this minimum. In this context, **the inherent structure approach allows us to explore**

[1] Polydispersity refers to the variation in particle sizes within a system, quantifying the degree of size diversity. This variation helps to prevent crystallization by disrupting uniform packing. Common models include bidisperse mixtures, which consist of two distinct particle types of different sizes, and polydisperse systems, where a continuous distribution of particle sizes is present.

the arrangement of particles in these amorphous energy minima by removing the influence of thermal fluctuations. Also dynamical observables, such as the displacement of particle i over a time interval t, can be computed in terms of transitions between inherent structures. In this approach, the displacement does not correspond to the actual trajectory taken by the particle but rather to the distance between its position in the inherent structure at time $t = 0$ and its position in the inherent structure at time t. This method focuses on capturing the system's movement between energy minima. However, some information is inevitably lost when studying inherent structures. In particular, **the dynamical information on short timescales**, such as the β-relaxation timescale, **is lost**.

Isoconfigurational Ensemble Another commonly used technique in numerical simulations is the so-called *Isoconfigurational Ensemble*. This method was developed to separate the thermal contribution from the structural contribution in dynamical observables [CG10, GJL$^+$07]. The technique involves running multiple MD simulations, all starting from the same equilibrated configuration of the fluid, but with initial velocities randomly assigned from the Maxwell-Boltzmann distribution at the corresponding temperature. For a given dynamical observable m, the value $m_i^\alpha(t)$ associated to particle i and run α is averaged over the n_α different runs:

$$\langle m_i(t)\rangle_{ISO} = \frac{1}{n_\alpha}\sum_{\alpha}^{n_\alpha} m_i^\alpha(t) \tag{2.17}$$

In particular when $m_i^\alpha(t) = \|\mathbf{x}_i^\alpha(t) - \mathbf{x}_i(0)\|$, one obtains a measure of the mobility of particle i averaged in the iso-configurational ensemble. This quantity is ubiquitous in numerical studies of glass-forming liquids and it is referred to as *Dynamical Propensity*. In theory, averaging over an infinite number of runs would completely eliminate the effects of initial velocities, isolating the dynamics purely determined by the structure of the initial configuration. In practice, averaging over a finite number of runs reduces the impact of velocities while still preserving the essential structural contributions to the observable. This approach ensures that the observable is less influenced by random thermal fluctuations.

While some information is inevitably lost when calculating iso-configurational averages, key features, especially dynamical heterogeneities, are retained [WCHF04]. This demonstrates that **dynamical heterogeneities are indeed rooted in the system's structure** (or energy landscape). As such, the iso-configurational mobility serves as a good proxy for a structural order parameter, with its correlations growing alongside the dynamical length scale.

However, a challenge with this technique is that the resulting value is difficult to interpret purely as a structural quantity. While it is indeed independent of the initial velocities and fully determined by the structure, it still requires the governing equations of motion (such as Newtonian or Langevin dynamics) to be defined and ran. This dependency on the dynamics introduces a layer of complexity, as the mobility reflects both the structural configuration and the specific rules governing particle motion.

The quest for a Static Correlation Length

In previous sections, we established that the only clearly increasing length scale in supercooled liquids is the dynamical one. However, to define an increasing amorphous order and

demonstrate the existence of a thermodynamic phase transition, we require a static correlation length. Numerous attempts have been made to identify such a length scale, with notable examples being the Random First-Order Transition (RFOT) theory and Point-to-set (PTS) correlation [BB04, MS06, CGV07, BCGV08]. The point-to-set (PTS) approach builds on a common method in statistical physics that exploits boundary conditions to detect growing static correlations. The key idea is that to observe the growth of amorphous order, one must measure how far the influence of amorphous boundary conditions extends into the system. Specifically, starting from an equilibrated configuration of a supercooled liquid, all particles outside a cavity of radius R are frozen, while the particles inside the cavity are allowed to evolve and equilibrate under the influence of the pinned boundary. These pinned particles impose an alignment with an amorphous structure.

To quantify this effect, one measures the overlap[2] distribution $P(Q)$ between the initial configuration and the equilibrium configurations within the cavity. Two scenarios arise: if the cavity is small, only one amorphous state is accessible, resulting in a high-overlap peak in $P(Q)$. Conversely, if the cavity is large enough, multiple amorphous states become accessible, and $P(Q)$ shows a peak at low overlap. Between these extremes, there is a critical cavity radius, $R \sim \xi_D$, where the distribution becomes bimodal, signaling a crossover between the two regimes. This crossover identifies the typical size of the correlated region and is a measure of the extent of the amorphous order.

While the study of point-to-set correlations has confirmed many predictions of RFOT theory and suggests a divergence of the static length scale at finite temperature (T_k) in three-dimensional models, there are limitations. First, the method is computationally demanding: finite-size cavities are not self-averaging, so it is necessary to repeat the overlap measurements for many independent quenched configurations and average over these disorder realizations. Additionally, running MD simulations inside the cavity is very slow and requires both SMC and parallel tempering. Second, the growth of the static length scale does not seem to match the dynamic one: the dramatic slowdown in dynamics is accompanied by only a modest increase in the static length scale [BJ12].

The last limitation highlights a key challenge in the study of supercooled liquids: identifying a structural length scale that grows as temperature decreases and can explain the expansion of dynamical domains, thereby linking static and dynamic correlations. In the following sections, we explore approaches that leverage Machine Learning techniques to define a structural order parameter and try to uncover this elusive length scale.

[2]Much similarly to the concept of replica overlap q introduced in the previous chapter, here the overlap Q between two configurations measures how similar they are. It can be computed as: $Q = \frac{1}{v}\sum_{i,j\in v}\Theta(a - |\mathbf{r}_i^{\mathcal{C}_0} - \mathbf{r}_j^{\mathcal{C}_1}|)$ where v refers to the volume of the cavity, and $\mathcal{C}_0$, $\mathcal{C}_1$ represent the initial and final configurations, respectively.

2.2 ML for Structural Glasses

As mentioned in the previous section, two major challenges remain in the development of a microscopic theory of glasses. The first is the **search for structural order**: specifically, identifying "defects" or locally preferred structures that could reveal short- or medium-range order within the glass. The second challenge involves **understanding the mechanisms driving glassy dynamics**, particularly the microscopic structural features that give rise to dynamical relaxation and heterogeneity. A key task here is to pinpoint "soft" and "hard" regions within the structure and establish a connection between structural characteristics and dynamical behavior. Both of these challenges present an ideal opportunity for the application of machine learning (ML) models, which are powerful tools for uncovering hidden patterns in complex datasets. Indeed, in recent years, a growing number of physicists have begun using ML techniques to tackle these problems.

For the first challenge, researchers have applied pattern-recognition techniques [Cos11, KIG11, RK17, TTR17, RTS20] and unsupervised learning approaches [BMAM$^+$20, PJC20, CJP22, RCH$^+$11, RCH$^+$12] to autonomously detect and classify structural motifs in supercooled liquids.

In addressing the second challenge, several studies have used expert, hand-crafted features, to describe the structure of super-cooled liquids which directly correlate with the dynamics [TT19, LBVP22], without the application of Machine Learning. Some approaches have leveraged unsupervised methods to extract relevant structural features from low-dimensional representations of liquid configurations [BMAM$^+$20, PJC20, CJP22]. The majority of recent solutions, though, rely on supervised learning techniques of varying complexity, aiming to predict particle mobility based on structural information [CSR$^+$15, BKGB$^+$20, BSF21, JBB23].

Ultimately, a comprehensive theory of the glass transition should integrate solutions from both these challenges. Some efforts in this direction have been discussed in the road-map paper [JAB$^+$23].

Here we examine key works addressing the second challenge, categorizing them into three groups: those that rely on expert-crafted features without the use of machine learning, those employing unsupervised learning approaches, and those utilizing supervised learning techniques.

2.2.1 Expert features

Tong et al. construct structural order parameters that detect sterically favored structures in instantaneous liquid states [TT19]. The method measures deviations from a perfectly packed arrangement. For example, in two-dimensional glass formers, for each particle, they consider all pairs of nearest neighbors, compute the angle formed by the three particles, and compare it to the angle they would form if they were in contact. The structural order parameter is then defined as:

$$\Theta_o = \frac{1}{N_o} \sum_{<ij>} |\theta_{ij}^{(1)} - \theta_{ij}^{(2)}| \tag{2.18}$$

where $\theta_{ij}^{(1)}$ is the actual angle, $\theta_{ij}^{(2)}$ is the ideal packing angle, and N_o is the number of pairs of neighbors. This order parameter is local and is computed based on instantaneous positions,

rather than inherent structures. By analyzing the macroscopic value of this parameter across the entire system, they discover a linear scaling relation with intensive thermodynamic variables that govern the transition to glassy states, such as temperature T and inverse density $1/\rho$. This relationship holds across different supercooled glass formers. At the macroscopic level, they establish a direct quantitative relationship between structural order parameters and relaxation time $\tau_\alpha(T, \rho)$. Furthermore, by spatially coarse-graining the system, they reveal a spatial correlation between maps of the structural order parameter and dynamical behavior. However their method achieves good precision only on hard-spheres models where the only interaction between particles is of excluded-volume type. Lennard-Jones models display a more complex relationship between structure and dynamics; in such models prediction of dynamical heterogeneites is intrinsically harder.

Lerbinger et al. demonstrate how local soft directions are key to understanding the rate of local relaxations in supercooled liquids [LBVP22]. Specifically, they construct local descriptors of softness by analyzing the inherent structures of the system and conducting local shear tests. In these tests, they impose a shear along a direction θ on a local region with a radius of five particle diameters using an athermal quasi-static deformation. They then compute the directional residual plastic strengths $\Delta\tau^c(\theta)$ and identify the softest direction by sampling multiple values of θ:

$$\Delta\tau^c_{min} = \min_\theta \Delta\tau^c(\theta) \tag{2.19}$$

This local, purely structural quantity correlates well with the spatial fluctuations of the coarse-grained dynamical propensity, revealing that soft directions strongly influence thermal relaxation. Moreover, they show that the vectorial thermal displacement between two IS overlaps significantly with the non-affine displacement caused by deformation along the weakest direction, further reinforcing the connection between local structural softness and dynamical behavior. Their methodology proves effective in studying the rheology of structural glasses as well [PVF16, RPS+20]. However, since their approach is rooted in inherent structures, it shows good correlation on the α-relaxation timescale while reduced accuracy on shorter times.

2.2.2 Unsupervised Learning

In unsupervised settings, the structure of a supercooled liquid is typically described using structural descriptors that capture the local environment of each particle. Dimensionality reduction algorithms are then applied to extract relevant features. This is often followed by clustering techniques to identify heterogeneous populations of particles, or the low-dimensional representations may be directly used to predict dynamical observables. A key debate in the field centers on whether supervised learning approaches, which fit dynamical measures such as propensity, simply replicate the dynamical data without uncovering meaningful structural properties. Unsupervised methods, in contrast, rely solely on structural information, avoiding this potential pitfall. However, while unsupervised approaches are generally effective at identifying structural heterogeneity, they are typically less successful at predicting dynamics compared to supervised methods.

Coslovich et al. explore dimensionality reduction methods to characterize the structure of supercooled liquids and aim to establish a connection between low-dimensional representations of

structural descriptors and dynamical fluctuations [CJP22]. Their focus is primarily on structural descriptors such as smoothed bond-order parameters (SBO) and the smooth overlap of atomic positions (SOAP). We highlight the definition of SOAP, as it provides a useful comparison with the present work. Within the SOAP descriptor the starting point is the density of neighbors around a central particle smeared by Gaussians of width σ:

$$\rho(\mathbf{r};i) = \sum_{j}^{N_b(i)} \exp\left(-\frac{|\mathbf{r}-\mathbf{r}_{ij}|^2}{2\sigma^2}\right) \tag{2.20}$$

This density is then expanded in a basis that incorporates both radial dependence via radial basis functions $g_n(r)$ and angular dependence through spherical harmonics $Y_{lm}(\hat{\mathbf{r}})$:

$$\rho(\mathbf{r};i) = \sum_{n=1}^{n_{\max}} \sum_{l=0}^{l_{\max}} \sum_{m=-l}^{-l} c_{nlm}(i)\, g_n(r) Y_{lm}(\hat{\mathbf{r}}), \tag{2.21}$$

the coefficients of the expansion are obtained as:

$$c_{nlm}(i) = \int d\mathbf{r}\rho(\mathbf{r};i) g_n(r) Y^*_{lm}(\hat{\mathbf{r}}). \tag{2.22}$$

The basic SOAP descriptor is then defined by the power spectrum $p_{nl}(i) = \sum_{m=-l}^{l} c^*_{nlm}(i) c_{nlm}(i)$. However, this quantity lacks sensitivity to angular correlations between particles at different distances. To address this, they introduce interactions between neighboring shells:

$$Q_{nn'l}(i) = \left(\frac{8\pi^2}{2l+1}\right)^{1/2} \sum_{m=-l}^{l} c^*_{nlm}(i) c_{n'lm}(i). \tag{2.23}$$

The full SOAP descriptor of a particle i is then represented by the feature tensor:

$$(\ldots, Q_{nn'l}(i), \ldots), \tag{2.24}$$

They apply both Principal Component Analysis (PCA) and Neural Auto Encoder (NEA) techniques to these descriptors, observing similar outcomes: while these methods effectively capture structural heterogeneity in glass-forming liquids, the principal components exhibit weak correlation with dynamical observables. As a result, they turn to supervised learning, performing linear regression on SOAP descriptors, which demonstrates a strong correlation with dynamic propensity.

Boattini et al. follow a procedure conceptually similar to that of Coslovich *et al.* They use bond-order parameters (BOP) to encode the local environments of the particles and implement a neural network auto encoder to perform dimensionality reduction [BMAM$^+$20]. This is followed by Gaussian mixture clustering, which identifies two clusters: one representing mobile particles and the other static particles. They define their order parameter as the probability P_{red} that a particle belongs to the mobile cluster. After performing spatial coarse-graining, they observe a strong correlation with dynamics in hard-sphere models and the Wahnström mixture, although the method performs less effectively on the Kob-Andersen mixture. At the macroscopic level, their order parameter exhibits an exponential relationship with the structural relaxation time τ_α. The BOP is similar to SOAP but contains less information, as it projects the density of

surrounding particles onto a sphere of radius 1, without using a radial basis. The BOP is defined as:

$$q_l(i) = \sqrt{\frac{4\pi}{2l+1} \sum_{m=-l}^{l} |q_{lm}(i)^2|} \tag{2.25}$$

$$q_{lm}(i) = \frac{1}{N_b(i)} \sum_{j \in \mathcal{N}_b(i)} Y_{lm}(\mathbf{r}_{ij}) \tag{2.26}$$

It's worth noting that not only the dynamical measure but also the structural parameter (the BOP) is averaged over neighboring particles in this work. Despite this, the quality of the correlation is lower than that observed in the results of Coslovich *et al.* demonstrating that some information is missing in this description.

2.2.3 Supervised Learning

The majority of works that exploit ML tools to establish a connection between a growing static order and a dynamic length-scale relies on supervised learning approaches. The first significant contributions in this area can be traced back to the pioneering works of *Liu et al.* [SLRR14, CSR+15, SCKL16, SCS+16]. In their studies, simple structural descriptors of the local environment around each particle were used as input for linear machine learning models, such as Support Vector Machines, to predict particle mobility. In these approaches, each particle is treated as an independent sample for the ML model, meaning that no direct interaction between the structural descriptors of different particles is considered. It's worth highlighting that this straightforward yet effective methodology has led to a number of physics-driven studies [STC+18, CIS+17, SSCL17, LBD+20, TRL22, ZXY+22], which have provided physical interpretations of the model's output, often referred to as *Softness*. More recent works have employed advanced techniques and more sophisticated structural descriptors. While these approaches sometimes result in less interpretable models, they allow to establish stronger correlations between structural and dynamical fluctuations.

Bapst et al. were the first to employ Graph Neural Networks to predict dynamic propensity (the iso-configurational average displacement of a particle) from the static structure of glass-forming liquids [BKGB+20]. Specifically, they focused on the 80:20 3D Kob-Andersen mixture. Instead of manually crafting structural descriptors, they allowed the GNN to learn them automatically. The only inputs provided to the network were the particle types in the mixture and their relative positions. GNNs are particularly well-suited for this task because they aggregate information over increasingly larger length scales as the number of "convolutional" layers increases, making them ideal for capturing long-range structural patterns in the material. Additionally, GNNs allow interactions between the descriptors of different particles, enhancing the predictive capacity.

This approach significantly improved the accuracy of predicting particle dynamics from static observables. The model exhibited a growing machine-learned length scale and demonstrated strong generalization to temperatures not seen during training, suggesting that the network was capturing meaningful features rather than overfitting the dataset. However, there are notable shortcomings: GNNs are complex models with a large number of parameters, which rely heavily on MLPs to update features. This complexity makes the model difficult to interpret, obscuring the specific physical features that drive the predictions of particle mobility.

Shiba et al. improved results obtained by Bapst *et al.* through the introduction of an auxiliary task of edge regression. To enhance the accuracy of mobility predictions, they regressed the change in particles' relative positions. This inevitably leads to an increase in number of parameters for the model though [SHSS23].

Boattini et al. mimicked the coarse-graining effectively produced by GNNs and were able to match the predictive performance of [BKGB+20] using a much simpler, more interpretable model [BSF21]. Their approach is more expert features oriented, relying on carefully crafted descriptors of the local environment around each particle. These descriptors capture both isotropic (distance-based) and angular information about the surrounding particles. The radial descriptors are defined as:

$$G_i^{(0)}(r,\delta,s) = \sum_{j\neq i: s_j=s} e^{-\frac{(r_{ij}-r)^2}{2\delta^2}} \tag{2.27}$$

which counts the number of neighbors J within the shell centered at distance r of width δ. The angular descriptors are given by:

$$q_i^{(0)}(l,m,r,\delta) = \frac{1}{Z}\sum_{j\neq i} e^{-\frac{(r_{ij}-r)^2}{2\delta^2}} Y_l^m(\mathbf{r}_{ij}), \tag{2.28}$$

where Z is a normalization constant and Y_l^m are the Spherical Harmonics as defined above. The rotationally invariant version of these angular descriptors is obtained by taking the norm:

$$q_i^{(0)}(l,r,\delta) = \sqrt{\frac{4\pi}{2l+1}\sum_{m=-l}^{l} |q_i^{(0)}(l,m,r,\delta)|^2}. \tag{2.29}$$

The complete feature vector describing the environment of particle i then consists of the values of $G_i^{(0)}(r,\delta,s)$ and $q_i^{(0)}(l,r,\delta)$.

They perform spatial averaging of these descriptors and feed their value averaged on multiple shells to a ridge regression model (linear model). Through this simpler and more interpretable architecture, they achieve prediction accuracy comparable to that of Bapst *et al.*, demonstrating the power of expert-driven features in glassy systems.

In a similar vein, Jung *et al.* and Alkemade *et al.* explored the impact of various physical observables that can be computed from the structure of a supercooled liquid on the quality of dynamical predictions [JBB23, ASF23].

Jung et al. employed an MLP with physics-informed input features. They highlighted the importance of several key factors in predicting particle mobility, such as the local potential energy (computed as the sum of pairwise interactions with neighbors), the perimeter of the Voronoi cell surrounding each particle, and the variance of potential energy. Their feature set includes the descriptors from Boattini *et al.* [BSF21] as well as these additional physical observables. They also applied spatial coarse-graining to these features. To further enhance their model, Jung *et al.* modified the typical mean squared error loss function by introducing additional terms that penalize deviations from the true variance and the true spatial correlations of the dynamic propensities (the target variable). This allowed them to achieve state-of-the-art results, with strong generalization across different temperatures and physical systems, while using far fewer

parameters than the GNN model from Bapst *et al.*, as they employed a simpler MLP architecture. Notably, their model precisely captures the growing length scale of dynamic propensity, although they achieve this by explicitly including this information in their loss function. While it is unsurprising that their model performs well in capturing spatial correlations when trained to do so, the fact that it generalizes effectively to other temperatures and systems suggests it is learning meaningful representations. A more detailed comparison with their results will be discussed in Sec. 2.4.3.

Alkemade et al. adopted the same descriptors introduced by Boattini *et al.* and employed a similar methodology, involving spatial averaging and the use of a simple linear model. However, they significantly improved the prediction quality of dynamic propensity on short-to-medium timescales by incorporating new information into the model, specifically the cage size and the interaction forces between neighboring particles. They developed an algorithm to estimate the cage size around each particle, which is purely a structural quantity as it depends solely on the initial particle positions. Their findings demonstrate that interaction forces are crucial for predicting particle mobility at short timescales (ballistic motion before being trapped in the cage), while cage size becomes the dominant factor at medium timescales (when the plateau of MSD starts to form), and the traditional structural descriptors become more relevant at long timescales, approaching the α relaxation time. Furthermore, they revealed that Inherent Structures lose all relevant dynamical information at short timescales but retain valuable insights at longer timescales. Some of these conclusions are corroborated by our own analysis, which will be discussed in detail in later sections.

Limitations

Works that exploit advanced machine learning models, such as GNNs [BKGB$^+$20, SHSS22], often suffer from two main shortcomings: a large number of parameters and a lack of interpretability. On the other hand, approaches based on expert-designed features frequently lack expressivity, as they typically use shallow models, and the chosen features may not be sufficiently informative to capture the structural factors driving particle dynamics.

Our work seeks to bridge this gap by leveraging the recent concept of group-equivariant machine learning [CW16a]. **We implement a roto-translational equivariant graph neural network to predict particle mobilities from their static positions**. This approach reduces the number of parameters and improves interpretability compared to standard GNNs. At the same time, we draw on the lesson learned from computer vision and the advent of CNNs, which highlights that meta-design (allowing the model to learn features with an expert-designed inductive bias) tends to outperform manual feature design. Expert-crafted features, while often invariant, can be less expressive than features learned by a neural network with a well-designed architecture.

By enforcing equivariance with respect to rotations and translations in our network, we allow for greater expressivity compared to simply invariant features, since invariance imposes a much stricter constraint than equivariance. Our model is designed to be SE(3)-equivariant, meaning that the hidden representations within the GNN are sensitive to translations and rotations. Concretely, under a rotation of the entire glass, scalar properties of particles (such as mobility) remain unchanged, while vectorial quantities (such as relative positions) transform accordingly.

With SE(3)-equivariant networks, internal representations behave like physical vectors, ensuring that the model's learned features rotate appropriately when the input undergoes rotation, providing both expressivity and interpretability.

In the next section we will define equivariant neural networks and all the theoretical concepts necessary to build them.

2.3 Equivariant Neural Networks

One of the key factors behind the remarkable success of deep learning in recent years is the integration of inductive biases directly into neural network architectures. This involves constraining the network to learn features that align with certain inherent properties of the data, such as symmetries in the system being modeled. For example, Convolutional Neural Networks (CNNs) achieved widespread success due to their inherent translation equivariance: a translation in the input produces a corresponding translation in the output. This property allows CNNs to efficiently capture spatial patterns, reflecting a fundamental symmetry in many tasks such as image recognition. Incorporating symmetries into neural networks helps in learning representations that respect the physical properties of the system under study, while also reducing the number of model parameters. This not only improves interpretability but also makes models more data-efficient.

While CNNs are naturally translation equivariant, many tasks involve more complex symmetries, such as rotations or scale changes. Simply enforcing invariance to these transformations, *i.e.* ensuring that a transformation in the input leaves the output unchanged, can be too restrictive, as it prevents the model from distinguishing between different orientations or scales of the same feature, leading to a loss of valuable information. This has led to a shift in focus from invariance to equivariance in many applications. Equivariance, as opposed to invariance, preserves the structure of the transformation, allowing the network to learn how different features behave under the same group of transformations. For example, in 3D tasks such as molecular modeling or object recognition, rotational equivariance is often crucial, as vectorial properties like magnetic moments must rotate in accordance with object orientation [FWFW20, BHvdP+21, LNC22, SHW22, LWDS23]. Beside this, in recent years, equivariant architectures have achieved competitive performance across various fields. These include simulation of fluid dynamics [TGB+23a, TGB+23b], approximation of inter-atomic potentials for ab-initio simulations [BMS+22, BKS+22, MBJ+23, GLZ+23], drug discovery and design [SGP+22, IAS23, SHD+24, LLC+24] and protein structure prediction with AlphaFold, winner of the Nobel Prize in Chemistry, being the most notable example [JEP+21, Nob24].

In this section, we review the theoretical foundations of equivariant neural networks. We begin by introducing the necessary mathematical background in Sec. 2.3.1, followed by the concept of group convolutions, which achieve equivariance with respect to roto-translations in Convolutional Neural Networks, discussed in Sec. 2.3.2. Next, in Sec. 2.3.3, we explore steerable convolutions, which allow for true equivariance without discretization by operating in the Fourier space. We then extend these concepts to GNNs, where convolutions are applied to point clouds that do not necessarily lie on a regular grid, as in CNNs (Sec. 2.3.4). Finally, we define the typical layer of an SE(3)-equivariant GNN, which serves as the core model in our work.

2.3.1 Mathematical Background

Here we review some elements of group theory, covering the notions of group structure, representations and equivariance. Two excellent references that explore these concepts and apply them to the context of equivariant networks are [Bek21] and [Vee24], which have served as inspiration for this section.

We start by defining a Group:

Definition 2.3.1: Group

A group G is a set, together with a binary operation $\cdot$, that combines two elements $g, h \in G$ to form another element $g \cdot h \in G$. The set and operation must satisfy four axioms:

1. **Closure**: For all $g, h \in G$, $g \cdot h \in G$.
2. **Associativity**: For all $g, h, k \in G$, $(g \cdot h) \cdot k = g \cdot (h \cdot k)$.
3. **Identity Element**: There exists an element $e \in G$ such that for all $g \in G$, $e \cdot g = g \cdot e = g$.
4. **Inverse Element**: For each $g \in G$, there exists an element $g^{-1} \in G$ such that $g \cdot g^{-1} = g^{-1} \cdot g = e$.

Two examples of group particularly relevant for our work are the Translation Group and the Special Orthogonal Group:

2.3.1 Example: *The translation group $(\mathbb{R}^n, +)$ consists of the set of translation vectors $\{\mathbf{x} \in \mathbb{R}^n\}$ equipped with group product and inverse given by:*

$$g \cdot h = \mathbf{x}_g + \mathbf{x}_h \tag{2.30}$$

$$g^{-1} = -\mathbf{x}_g \tag{2.31}$$

2.3.2 Example: *The Special Orthogonal Group, denoted as $SO(n)$, describes the group of continuous rotations in an n-dimensional space. These rotations are parameterized by a set of angles Θ, with the number of required angles depending on the dimensionality of the space. Notable examples include $SO(2)$, representing planar rotations, which requires a single angle, and $SO(3)$, representing volumetric rotations, which requires three angles. The action of a rotation $r_\Theta \in SO(n)$ is defined as:*

$$\forall x \in X = \mathbb{R}^n : r_\Theta, x \mapsto r_\Theta \cdot x = \psi(\Theta)x$$

Where $\psi(\Theta)$ is the rotation matrix, i.e., for rotations in $SO(2)$ with $\Theta = \{\theta\}$:

$$\psi(\Theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix},$$

and $\psi(\Theta)x$ is the regular matrix-vector product.
The $SO(n)$ groups are therefore the group of all $n \times n$ orthogonal real matrices with positive determinant:

$$SO(n) = \left\{ O \in \mathbb{R}^{n \times n} \mid O^\top O = I_n \text{ and } \det(O) = 1 \right\}.$$

equipped with group product and inverse given by the matrix product and the matrix inversion operation.

In previous example we exploited the concept of group action, let's formalize it:

Definition 2.3.2: Group Action

The action of group G on a space X is a binary operator $\odot$: $G \times X \to X$ that follows the group structure:

$$g \odot (\tilde{g} \odot x) = (g \cdot \tilde{g}) \odot x$$

with $g, \tilde{g} \in G$ and $x \in X$.

in particular, if for any two points $x, y \in X$ there exists an element $g \in G$ such that the action of g moves x to y the space X is called a *Homogeneous Space* of G. It is worth noting that X can also be a group itself, allowing us to define the action of one group on another in a similar manner.

A relevant concept in defining the roto-translation group, with respect to which we aim to enforce equivariance in our network, is the Semi-Direct Product. It is defined as follows:

Definition 2.3.3: Semi-Direct Product

Consider two groups $(N, *)$ and $(H, +)$, along with a group action $\phi : H \times N \to N$ of H on N. The *semi-direct product* group $N \rtimes_\phi H$ is defined as the Cartesian product $N \times H$ with the following group operation:

$$(n_1, h_1) \cdot (n_2, h_2) = (n_1 * \phi(h_1, n_2), h_1 + h_2)\,.$$

As in a standard Cartesian product, each element of the semi-direct product can be uniquely identified by a pair of elements from the two subgroups. We can now look at the Special Euclidean Group *i.e.* the group of roto-translations in n dimensions:

2.3.3 Example: *The Special Euclidean Group, denoted as $SE(n)$, is the group of all translations and rotations in n-dimensional space. This group can be represented as the semi-direct product of the translation group $(\mathbb{R}^n, +)$ and the special orthogonal group $SO(n)$, which governs rotations.*
Any element of $SE(n)$ can be identified as a pair (t_v, r_Θ), where $t_v \in \mathbb{R}^n$ represents a translation vector and $r_\Theta \in SO(n)$ is a rotation. The group operation between two elements (t_{v_1}, r_{Θ_1}) and (t_{v_2}, r_{Θ_2}) is given by:

$$(t_{v_1}, r_{\Theta_1}) \cdot (t_{v_2}, r_{\Theta_2}) = (t_{v_1} + r_{\Theta_1} t_{v_2}, r_{\Theta_1} r_{\Theta_2}).$$

In this operation, the translation vectors are composed by applying the rotation r_{Θ_1} to the second translation t_{v_2} before adding it to the first translation t_{v_1}, while the rotations are composed as usual in the special orthogonal group.

Next, we introduce the concept of a linear representation, which is valuable because it maps group elements to matrices that act on vector spaces. This allows us to practically handle operators representing group elements. In fact, we applied this concept informally when we introduced the rotation group, describing it as the group of orthogonal matrices. In doing so, we were actually working with a linear representation of the group $SO(n)$. Now, let's formally

define a linear group representation:

Definition 2.3.4: Linear Group Representation

A Linear Group Representation ρ of a group G on a vector space V is a group homomorphism from G to the general linear group $GL(V)$, i.e., it is a map:

$$\rho : G \to GL(V) \qquad s.t. \qquad \forall g_1, g_2 \in G \qquad \rho(g_1 \cdot g_2) = \rho(g_1)\rho(g_2)$$

if $V \equiv \mathbb{R}^3$, $GL(V)$ corresponds to the group of three dimensional matrix with real entries.

Another important concept is that of the left-regular representation. This allows us to define the action of group elements on a function by transforming the function's domain, i.e.:

Definition 2.3.5: Left-regular representation

Let $f \in \mathbb{L}_2(X)$ and $\odot$ denote the action of the group G on the domain of X. Then the left-regular representation of G acting on $\mathbb{L}_2(X)$ is given by:

$$[\mathcal{L}_g f](x) = f(g^{-1} \odot x)$$

The concept becomes intuitive if we consider an example where we have a function $f(\mathbf{x})$ with $\mathbf{x} \in \mathbb{R}^2$, and we want to apply a rotation $r_\theta \in SO(2)$. The left-regular representation tells us that we should rotate the domain of the function in the opposite direction, i.e.:

$$[\mathcal{L}_{r_\theta} f](\mathbf{x}) = f(r_{-\theta}\mathbf{x}) \tag{2.32}$$

When working with matrix representations of groups, a key concept is that of *irreducible representations* (irreps), which enable the decomposition of a representation into simpler components, much like a Fourier decomposition. Suppose a group G acts on a vector space V, which can be decomposed into two invariant sub-spaces V_1 and V_2, such that $V = V_1 \oplus V_2$. Invariance under the action of $\rho(G)$ means that for all $g \in G$, $\rho(g)V_i \subseteq V_i$. In this case, the matrix of $\rho(g)$ can be expressed in block-diagonal form:

$$\rho(g) = \begin{bmatrix} \rho_1(g) & 0 \\ 0 & \rho_2(g) \end{bmatrix} \tag{2.33}$$

where $\rho_1 : G \to \mathrm{GL}(V_1)$ and $\rho_2 : G \to \mathrm{GL}(V_2)$. Thus, the representation decomposes as the direct sum of the two representations acting on the sub-spaces: $\rho(g) = \rho_1(g) \oplus \rho_2(g)$. In principle, this factorization can continue, breaking down ρ_1 and ρ_2 further until we reach irreducible representations.

Definition 2.3.6: Irreducible Representation

A representation is irreducible if it does not contain any non-trivial invariant sub-spaces, meaning it cannot be further decomposed.

An important property of irreducible representations is that, for the types of groups and vector spaces of interest in this work, the Peter-Weyl theorem applies. This theorem states that any

representation can be decomposed into a direct sum of irreducible representations through an appropriate change of basis Q:

$$\rho(g) = Q\left[\bigoplus_{i\in I}\psi_i(g)\right]Q^{-1} \tag{2.34}$$

where I is the index set of irreducible representations of ρ.

Moreover, this result extends to square-integrable functions defined on the group G i.e. functions belonging to $\mathbb{L}^2(G)$. Specifically, the matrix coefficients of the irreducible representations of G form a spanning set for the vector space $\mathbb{L}^2(G)$. An explicit orthonormal basis for the space of complex-valued square-integrable functions in $\mathbb{L}^2(G)$ can be constructed as:

$$\left\{\sqrt{d_\psi}[\psi(g)]_{ij} \;\middle|\; \psi \in \hat{G}, 1 \le i,j \le d_\psi\right\} \tag{2.35}$$

where $\hat{G}$ denotes the set of irreducible representations of G and d_ψ is the dimension of the irreducible representation ψ.

Since irreps form an orthonormal basis of $\mathbb{L}^2(G)$, they can be used to express a function $f : G \to \mathbb{C}$ trough the *inverse Fourier transform*:

Definition 2.3.7: Inverse Fourier Transform

Let G be a group and $f : G \to \mathbb{C}$ a function defined on the group. The inverse Fourier transform allows us to express f as a linear combination of the irreps of G, namely:

$$f(g) = \sum_{\psi_j\in\hat{G}} \sqrt{d_j}\,\mathrm{Tr}\left(\psi_j(g)^T \hat{f}(\psi_j)\right)$$

$\hat{f}(\psi_j)$ denotes the Fourier coefficient corresponding to the irrep ψ_j and it is computed as follows:

$$\hat{f}(\psi_j) = \sqrt{d_j}\int_G f(g)\psi_j(g)\,\mathrm{d}g$$

Having established how to represent functions on groups using irreps, the final tool we need is a method for combining different irreps. This is accomplished through the Clebsch-Gordan tensor product:

Definition 2.3.8: Clebsh-Gordan Tensor Product

Let G be a compact group and let $\rho_l : G \to GL(V_l)$ and $\rho_k : G \to GL(V_k)$ be two irreducible representations of G acting on vector spaces V_l and V_k, respectively. The *tensor product representation* $\rho_l \otimes \rho_k$ is defined as the representation acting on the tensor product space $V_l \otimes V_k$ given by:

$$(\rho_l \otimes \rho_k)(g) = \rho_l(g) \otimes \rho_k(g)$$

If $|V_l| = n_l$ and $|V_k| = n_k$ then $\rho_l \otimes \rho_k \in \mathbb{C}^{n_l n_k \times n_l n_k}$ assuming complex valued representations. This representation is not necessarily irreducible. The Clebsch-Gordan decomposition states that the tensor product of two irreducible representations is, in general, reducible and can be decomposed into a direct sum of irreducible representations:

$$(\rho_l \otimes \rho_k)(g) = \left[C^{lk}\right]^\top \left(\bigoplus_j \bigoplus_s^{[j(lk)]} \psi_j(g) \right) C^{lk}$$

where $\psi_j : G \to GL(V_j)$ are the irreps necessary for the decomposition, each of them with a multiplicity $[j(lk)]$. $C^{lk} \in \mathbb{C}^{n_l n_k \times n_l n_k}$ is the matrix of change of basis. We can identify block of columns in C^{lk} such that each irreps acts separately on them by defining $C^{lk}_j \in \mathbb{C}^{n_j \times n_l n_k}$. In this way we can rewrite the decomposition as a sum over irreps:

$$(\rho_l \otimes \rho_k)(g) = \sum_j \sum_s^{[j(lk)]} [C^{lk}_{j,s}]^T \psi_j(g) C^{lk}_{j,s}$$

The index s in the coefficients is introduced to account for the multiplicity of irreps. C^{lk}_j are called *Clebsch-Gordan coefficients* and allow the projection on the subspace associated to irrep j of the tensor product of irreps l and k.

We conclude this section by stating the definition of *Equivariance* which is our ultimate goal:

Definition 2.3.9: Equivariance

An operator $\phi : X \to Y$ which maps elements from an input space X to an output space Y is equivariant with respect to a group G if and only if:

$$\forall g \in G : \qquad \rho_X(g) \circ \phi = \phi \circ \rho_Y(g)$$

This means that if an operator is equivariant with respect to the action of a group, upon transformation of the input, the output transforms accordingly. A straightforward example is the convolution operator which is equivariant with respect to translations: if the input field is translated, the output field is translated accordingly. In the following, Φ will represent one parametrized layer of the neural network and the goal is to design its action in such a way that it fulfills equivariance with respect to a group G.

2.3.2 Group Convolutions

Group convolution is a generalization of the standard convolution operation used in CNNs, designed to implement equivariance with respect to a group G [CW16a]. We begin by revisiting the concept of regular convolution, demonstrating its inherent translation equivariance. Building on this, we extend the idea to a general group G, deriving the formula for group convolution. However, there are certain limitations when implementing practically group convolution, which we will explore in the final part.

In a standard CNN, the input signal is typically[3] a function $f : \mathbb{R}^n \to \mathbb{R}$ and the kernel is another function $k : \mathbb{R}^n \to \mathbb{R}$. The action of a convolutional layer is defined by the *regular convolution* operation (or more precisely, cross-correlation):

$$[k * f](x) = \int_{\mathbb{R}^n} k(\tilde{x} - x) f(\tilde{x}) \, \mathrm{d}\tilde{x} \tag{2.36}$$

This amounts to sliding the kernel over all the possible translations $x \in \mathbb{R}^n$ and for each translation compute the inner product of $\mathbb{L}_2(\mathbb{R}^n)$ defined as:

$$\langle f, k \rangle_{\mathbb{L}_2(\mathbb{R}^n)} = \int_{\mathbb{R}^n} f(\tilde{x}) k(\tilde{x}) \, \mathrm{d}\tilde{x} \tag{2.37}$$

This inner product serves as a similarity measure, meaning the convolution essentially computes the similarity between the kernel and the input signal at each point in space. Now, considering the group of translations $(\mathbb{R}^n, +)$ and exploiting the definition of left-regular representation (2.3.1) we can express the regular convolution as:

$$[k * f](x) = \langle \mathcal{L}_{x \in \mathbb{R}^n} k, f \rangle_{\mathbb{L}_2(\mathbb{R}^n)} \tag{2.38}$$

It is straightforward to show that this operation is equivariant to translations, as a translation of the input signal can be handled by a simple change of variables. This leads to the following equivariance relation:

$$[k * \mathcal{L}_{t \in \mathbb{R}^n} f](x) = \mathcal{L}_{t \in \mathbb{R}^n} [k * f](x) \tag{2.39}$$

Thus, a translation applied to the input signal is reflected equivalently in the output of the convolution.

This concept can be generalized by replacing the action of the translation group with a more general group G. This leads to the definition of *group convolution*, which is equivariant to the action of G.

[3] In practice, when dealing with 2D images, $n = 2$ and for colored images, there are usually 3 channels corresponding to RGB values. Thus, the common setup is $f : \mathbb{R}^2 \to \mathbb{R}^3$, with the kernel often being a matrix that maps a given number of input channels to an arbitrary number of output channels. For simplicity, we consider the case of a single channel here, though the derivation can be easily extended to multiple channels.

Definition 2.3.10: Group Convolution

Given a group G, a signal $f : G \to \mathbb{R}$ and a kernel $k : G \to \mathbb{R}$ the group convolution is defined as:

$$\begin{aligned}[k *_G f](g) &= \int_G k(g^{-1}\tilde{g}) f(\tilde{g}) \, \mathrm{d}\tilde{g} \\ &= \langle \mathcal{L}_{g \in G} k, f \rangle_{\mathbb{L}_2(G)}\end{aligned}$$

Equivariance can be proved by applying a group element $h \in G$ to the input signal:

$$\begin{aligned}[k *_G \mathcal{L}_h f](g) &= \int_G k(g^{-1}\tilde{g}) f(h^{-1}\tilde{g}) \, \mathrm{d}\tilde{g} && (2.40) \\ &\text{change of variable } \tilde{g} = h^{-1}\tilde{g} \\ &= \int_G k(g^{-1}(h\tilde{g})) f(\tilde{g}) \, \mathrm{d}\tilde{g} \\ &= \int_G k((h^{-1}g)^{-1}\tilde{g}) f(\tilde{g}) \, \mathrm{d}\tilde{g} = \mathcal{L}_h[k *_G f](g) && (2.41)\end{aligned}$$

with $\mathrm{d}\tilde{g}$ the left Haar measure on G.

In practical applications, the group of interest for equivariant neural networks is commonly $SE(n) = (\mathbb{R}^n, +) \rtimes SO(n)$ the group of roto-translation in n dimension. In this case the group convolution takes the following form:

$$[k *_{SE(n)} f](x, r_\Theta) = \int_{\mathbb{R}^n} \int_{SO(n)} k(r_\Theta^{-1}(\tilde{x} - x), r_\Theta^{-1}\tilde{r}_\Theta) f(\tilde{x}, \tilde{r}_\Theta) \, \mathrm{d}\tilde{x} \, \mathrm{d}\tilde{r}_\Theta \tag{2.42}$$

This generalizes the concept of regular convolution in a natural way. Just as regular convolution achieves translation equivariance by applying all possible translations to the kernel, here, to achieve roto-translation equivariance, the kernel is both translated and rotated during convolution. However, there's an important caveat: the input signal must be defined on the group space $SE(n)$, whereas, in typical scenarios, the input signal is defined on $\mathbb{R}^n$. To address this, group equivariant neural networks introduce a first layer called the *lifting convolution*, which lifts the input signal from $\mathbb{R}^n$ to the space of the group. This lifting operation is performed as follows:

$$[k *^{lift}_{SE(n)} f](x, r_\Theta) = \int_{\mathbb{R}^n} k(r_\Theta^{-1}(\tilde{x} - x)) f(\tilde{x}) \, \mathrm{d}\tilde{x} \tag{2.43}$$

The lifting convolution is an equivariant operation as well.

Group convolutions, while theoretically effective, face certain **limitations in practical applications**. First, to lift the input function to the group space, one must discretize the possible rotations. This means that the resulting network will only be equivariant to a finite set of discrete rotations, rather than to continuous rotations, which may lead to a loss of precision [LBP+20]. Second, there is a significant computational challenge: sampling and applying discrete rotations at various finite angles is computationally expensive, especially as the dimensionality of the group increases. This leads to a high computational overhead, particularly in 3D tasks or when the group has complex structure, such as the full roto-translation group $SE(3)$.

2.3.3 Steerable Convolutions

In order to overcome limitations of group convolution CNNs, Cohen *et al.* exploited the Fourier representation introduced in definition 2.3.1 to introduce *Steerable Convolutions* [CW16b].

In group convolutions, features are defined as a field $f : G \to \mathbb{R}$ with $G = (\mathbb{R}^n, +) \rtimes H$. Thus the signal on which the convolution layer acts is a field $f(x, h)$ whose domain is extended, or lifted, to the space of the group G. In the example shown above $H = SO(n)$, here we adopt a more general approach considering a generic subgroup H.

An alternative approach involves defining $|H|$-dimensional feature vectors over the spatial domain, i.e., $f : \mathbb{R}^n \to \mathbb{R}^{|H|}$. In this framework, the input signal is a feature field $f(x) \in \mathbb{R}^{|H|}$, which assigns to each point in space a vector of features known as *group fibers*. These fibers reside in an H-dimensional space, and the action of a subgroup element $h \in H$ on them is described by its representation $\rho_H(h)$. Consequently, we must generalize the left-regular representation, as the entire group G acts on the feature field in a more complex way:

$$[\mathcal{L}_g f](x) = \rho_H(h) f(g^{-1}x) \tag{2.44}$$

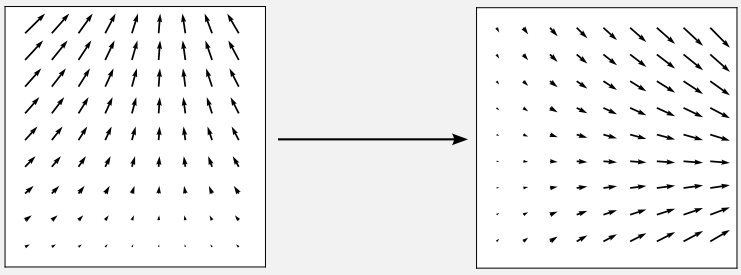
where the spatial domain is transformed by the full group G, while the fibers in the codomain (output domain) are transformed only by the subgroup H. This means that the group G applies transformations to the input's spatial coordinates, while the internal structure of the features (the fibers) is transformed by the subgroup H.

Let's explore a practical example:

2.3.4 Example: *Let $\mathbf{f} : \mathbb{R}^2 \to \mathbb{R}^2$ be a two-dimensional field defined on the plane and $G = SE(2) = \mathbb{R}^2 \rtimes SO(2)$ the group of roto-translation. An element $g = \mathbf{t}r_\theta$ acts on $\mathbf{f}$ as follows:*

$$[\mathcal{L}_g \mathbf{f}](x) = R(\theta)\mathbf{f}(R^{-1}(\theta)(\mathbf{x} - \mathbf{t}))$$

Roto-translating a vectorial field consists in applying the inverse roto-translation to its domain while the vectors are rotated by the angle θ as shown by the following sketch:



In particular one can choose to express this field in terms of irreps: given an irrep $\psi_j : H \to GL(\mathbb{R}^{d_j})$ we can define an *irrep field* $f_{\psi_j} : \mathbb{R}^n \to \mathbb{R}^{d_j}$ such that it is transformed according to the action of the irrep:

$$[\mathcal{L}_g f_{\psi_j}](x) = \psi_j(h) f_{\psi_j}(g^{-1}x) \tag{2.45}$$

One can concatenate multiple feature vectors belonging to different irreps space (with varying multiplicity), thanks to the direct sum the generalization is straightforward: let $f(x) = \bigoplus_j \bigoplus_s^{[j]} f_{\psi_j}(x)$, the action of the group is as follows:

$$[\mathcal{L}_g f](x) = \begin{bmatrix} \psi_{j_1}(h) & 0 & 0 & \cdots & 0 \\ 0 & \psi_{j_2}(h) & 0 & \cdots & 0 \\ 0 & 0 & \psi_{j_3}(h) & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \psi_{j_n}(h) \end{bmatrix} \begin{bmatrix} f_{\psi_{j_1}}(g^{-1}x) \\ f_{\psi_{j_2}}(g^{-1}x) \\ f_{\psi_{j_3}}(g^{-1}x) \\ \vdots \\ f_{\psi_{j_n}}(g^{-1}x) \end{bmatrix} \tag{2.46}$$

Consider a scenario where both the input and output fields are of the group fiber type. We aim to map the input field to the output field through a network layer, while ensuring that this mapping is equivariant with respect to the group G. As shown in [WGW+18], the most general equivariant linear maps are convolutions with H-steerable kernels.

Let $f_{\text{in}} : \mathbb{R}^n \to \mathbb{R}^{c_{\text{in}}}$ be the input field and $f_{\text{out}} : \mathbb{R}^n \to \mathbb{R}^{c_{\text{out}}}$ be the output field. The action of an element h from the subgroup H on these spaces is represented by $\rho_{\text{in}}(h)$ and $\rho_{\text{out}}(h)$, respectively. The condition for H-steerability of the kernel $k : \mathbb{R}^n \to \mathbb{R}^{c_{\text{out}} \times c_{\text{in}}}$ is given by:

$$k(hx) = \rho_{out}(h)k(x)\rho_{in}(h^{-1}) \quad \forall h \in H, x \in \mathbb{R}^n \tag{2.47}$$

The steerable convolution operation is then defined as:

$$f_{\text{out}}(x) = [k * f_{\text{in}}](x) = \int_{\mathbb{R}^n} k(\tilde{x} - x) \cdot f_{\text{in}}(\tilde{x})\, d\tilde{x} \tag{2.48}$$

This operation ensures that the mapping from the input to the output field respects the desired equivariance with respect to G.

To explicitly determine the form of the kernel, one must solve the constraint in Eq. 2.47. It can be simplified by expanding ρ_{in} and ρ_{out} in terms of irreps to factorize the constraint over independent sub-spaces:

$$k(hx) = Q_{\text{out}}^{-1} \left(\bigoplus_{i \in I_{\text{out}}} \psi_i(h) \right) Q_{\text{out}} k(x) Q_{\text{in}}^{-1} \left(\bigoplus_{j \in I_{\text{in}}} \psi_j(h)^{-1} \right) Q_{\text{in}} \quad \forall h \in H,\, x \in \mathbb{R}^n. \tag{2.49}$$

Next, we perform a change of variables, defining $\tilde{k} = Q_{\text{out}} k(x) Q_{\text{in}}^{-1}$, which gives:

$$\tilde{k}(hx) = \left(\bigoplus_{i \in I_{\text{out}}} \psi_i(h) \right) \tilde{k}(x) \left(\bigoplus_{j \in I_{\text{in}}} \psi_j(h)^{\top} \right) \quad \forall h \in H,\, x \in \mathbb{R}^n. \tag{2.50}$$

By exploiting the block diagonal structure of the irrep expansions, we can factorize the equation. We define $\tilde{k}^{ij} \in \mathbb{R}^{d_i \times d_j}$ as the diagonal block of $\tilde{k}$ corresponding to irreps $i \in I_{\text{out}}$ and $j \in I_{\text{in}}$. The constraint then reduces to:

$$\tilde{k}_{ij}(hx) = \psi_i(h) \tilde{k}_{ij}(x) \psi_j(h)^{\top} \quad \forall h \in H,\, x \in \mathbb{R}^n. \tag{2.51}$$

This formulation simplifies the problem by breaking the kernel into smaller, independent blocks based on the irreps, making it easier to handle and solve.

Explicit solution for SE(3) We show the explicit solution to Eq. 2.51 in the case of $G = (\mathbb{R}^3, +) \rtimes SO(3) = SE(3)$, which is the group of interest for our work. For a detailed derivation see the work of Weiler *et al.* in [WGW+18].

First, we need to introduce the irreps of $SO(3)$, known as Wigner-D matrices:

Definition 2.3.11: Irreps of SO(3)

Let $r_{(\alpha,\beta,\gamma)} \in SO(3)$ be an element of the rotation group in three dimensions, parameterized by three angles. There exists a family of matrix representations, indexed by their order $l \geq 0$, which act on vector spaces of dimension $|V_l| = 2l + 1$ and form the irreducible representations of $SO(3)$ on the subspace V_l. These matrices are known as Wigner-D matrices and are denoted by $D^l(r_{(\alpha,\beta,\gamma)})$. An important property of Wigner-D matrices is that their central column corresponds to spherical harmonics, i.e.,

$$D^l_{m,0}(r_{(\alpha,\beta,\gamma)}) = \sqrt{2l+1} Y^l_m(\mathbf{n}_{(\alpha,\beta)}) \tag{2.52}$$

where $Y^l_m(\mathbf{n}_{(\alpha,\beta)}) = N e^{im\varphi} P^m_l(\cos\theta)$ are the spherical harmonics, with $l \geq 0$, $m = -l, \ldots, l$, and $P^m_l(\cos\theta)$ are the associated Legendre polynomials.

Spherical harmonics form a basis of functions on the sphere $\mathcal{L}_2(\mathbb{S}^2)$ through Fourier representation, but they can also be interpreted as functions on $SO(3)$, invariant to rotations around the third angle γ. As part of the Wigner-D matrices, spherical harmonics are steerable functions. In next sections we will exploit spherical harmonics to transform input data, typically residing on the sphere, into steerable features, and then process these features through steerable convolutions.

Equation 2.51 can be re-formulated for $SO(3)$ in the following way:

$$\tilde{k}_{lj}(r_{(\alpha,\beta,\gamma)}x) = D^l(r_{(\alpha,\beta,\gamma)}) \tilde{k}_{lj}(x) D^j(r_{(\alpha,\beta,\gamma)})^\top \tag{2.53}$$

The solution is found in [WGW+18]:

$$\tilde{k}_{lj}(x) = \sum_{J,n} w^n_{lj,J}\, \tilde{k}^n_{lj,J} = \sum_{J,n} w^n_{lj,J}\, \mathrm{unvec}(\varphi^n(\|x\|)[C^{lj}_J]^T Y^J(x/\|x\|)) \tag{2.54}$$

Here, $w^n_{lj,J} \in \mathbb{R}$ represents the weights of a linear combination of basis functions for the kernel. The functions $\varphi^n(\|x\|)$ are the basis functions for the radial part of the kernel, which is not constrained by the H-steerability condition and is therefore free. The terms $C^{lj}_J \in \mathbb{C}^{(2J+1)\times(2l+1)(2j+1)}$ are the Clebsch-Gordan coefficients, $Y^l \in \mathbb{C}^{2l+1}$ is the array of spherical harmonics of rotational order l and the *unvec* operation restores the kernel to the correct shape of $(2l+1) \times (2j+1)$.

As mentioned earlier, we will work with features that belong to steerable spaces, so we can assume that $\tilde{k}_{lj}$ is the actual kernel we will be using (there is no need to rotate back to the initial representation). Thus, we can focus on specific irreps in both the input and output for the steerable convolution and compute the contribution of irrep j of the input field to irrep l of the output one:

$$f^l_{\mathrm{out},j}(x) = \int_{\mathbb{R}^n} \tilde{k}_{lj}(\tilde{x} - x) \cdot f^j_{\mathrm{in}}(\tilde{x})\, d\tilde{x} \tag{2.55}$$

Furthermore, due to the interaction with the kernel through the Clebsch-Gordan coefficients, multiple input irreps will contribute to the irrep l in output, then:

$$f^l_{\text{out}}(x) = \sum_{j \in I_{in}} \int_{\mathbb{R}^n} \tilde{k}_{lj}(\tilde{x} - x) \cdot f^j_{\text{in}}(\tilde{x})\, d\tilde{x} \tag{2.56}$$

By substituting the explicit form of the convolution kernel the steerable convolution on $SE(3)$ reads:

Definition 2.3.12: Steerable Convolution on $SE(3)$

$$f^l_{\text{out}}(x) = \int_{\mathbb{R}^n} \sum_{J,j,n} w^n_{lj,J}\, \varphi^n(\|\tilde{x} - x\|) Y^J((\tilde{x} - x)/\|\tilde{x} - x\|)) \otimes^{(lj)}_J f^j_{\text{in}}(\tilde{x})\, d\tilde{x}$$

where we have used the notation $\otimes^{lj}_J$ for the Clebsch-Gordan tensor product defined as follows: $h^J \otimes^{(lj)}_J h^j = h^J\, [CG]^{lj}_J\, h^j$ with $[CG]^{lj}_J \in \mathbb{C}^{(2l+1)\times(2J+1)\times(2j+1)}$ which represents the un-vectorization of the Clebsch-Gordan coefficients defined in 2.3.1. Notice that $[CG]^{lj}_J \neq 0$ only for $l \in [|j - J|, j + J]$.

2.3.4 SE(3) Steerable GNNs

The concept of steerable convolutions has been widely adopted in neural networks and, in particular, has been generalized to Graph Neural Networks, where the input field is typically composed of point-wise features in space [TSK+18, BMS+22, LS22, BHvdP+21]. In this section, we review the construction of a GNN and adapt the previous discussion on steerable convolutions to this setup.

Graph Neural Networks Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \dots, n_v\}$ is the set of vertices or nodes v_i and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges e_{ij}, respectively endowed with node features $\mathbf{h}_i \in \mathbb{R}^{c_v}$ and edge features $\mathbf{a}_{ij} \in \mathbb{R}^{c_e}$. GNNs operate on such graphs by updating node (and possibly edge) features through local operations on the neighborhood of each node. These operations are designed to adapt to different kinds of neighborhoods and respect node-index permutation equivariance, which are the two key features of GNNs, as opposed to CNNs (for which the learned kernels must have fixed, grid-like geometry, and for which each neighboring pixel is located at a fixed relative position). In this work we deal with Graph Convolutional Networks (GCN), a subclass of GNNs. A GCN layer acts on node features as follows:

$$\mathbf{h}'(\mathbf{x}_i) = \sum_{j \in \mathcal{N}(i)} \kappa(\mathbf{x}_j - \mathbf{x}_i)\mathbf{h}(\mathbf{x}_j) \tag{2.57}$$

where $\mathcal{N}(i)$ is the neighborhood of node i. Here a position $\mathbf{x}_i \in \mathbb{R}^3$ is associated to each node and κ is a continuous convolution kernel which only depends on relative nodes' positions. In this case, as for CNNs, the node update operation is translation-equivariant by construction. It is however not automatically rotation-equivariant.

Equivariance We can represent the point cloud processed by the GNN as a vector field $\mathbf{h}(\mathbf{x}) = \sum_{i \in \mathcal{V}} \delta(\mathbf{x} - \mathbf{x}_i)\mathbf{h}_i$ with values in some vector space H, and the action of a layer as a mapping $\mathcal{K}$ from one field $\mathbf{h}$ to the updated one $\mathbf{h}'$. We have seen in previous section that an element $t_v\, r_{(\alpha,\beta,\gamma)} \in SE(3)$ which is composed of a translation t_v and a rotation $r_{(\alpha,\beta,\gamma)}$, will act on the vector field as follows:

$$\mathbf{h}(\mathbf{x}) \xrightarrow{\rho(t_v\, r_{(\alpha,\beta,\gamma)})} \rho_H(r_{(\alpha,\beta,\gamma)})\mathbf{h}(\mathbf{R}^{-1}(\alpha,\beta,\gamma)(\mathbf{x}-\mathbf{v})) \tag{2.58}$$

The codomain is transformed by the representation of the rotation while the domain is transformed by the one of the inverse roto-translation. The definition of equivariance in this set-up translates as follows: let there be a mapping $\mathcal{K} : \mathbf{h}(\mathbf{x}) \to \mathbf{h}'(\mathbf{x})$ and $\mathbf{h} \in H$, $\mathbf{h}' \in H'$ with H, H' two vector spaces. The kernel $\mathcal{K}$ is equivariant with respect to G if

$$\forall g \in G \;\; \mathcal{K} \circ \rho(g) = \rho'(g) \circ \mathcal{K} \tag{2.59}$$

with ρ, ρ' representations of G on H and H' respectively. The input and output codomains H, H' do not need to be identical, and this is taken into account by the group representations $\rho_H(g)$ and $\rho_{H'}(g)$. A direct consequence of this definition is that invariance is a particular case of equivariance where $\rho_{H'}(g) = \mathbb{I} \;\; \forall g \in G$.

When dealing with $SE(3)$, the *only* invariant quantities are *scalars*, thus considering only invariant features would significantly reduce the model's expressivity.

Equivariant Features In the context of steerable networks, the input signals are represented in a *steerable basis*. In particular, we have seen in definition 2.3.3 that spherical harmonics form a steerable basis of complex functions on the sphere. Here we limit ourselves to real spherical harmonics $Y_m^l : \mathbb{S}^2 \to \mathbb{R}$. They can be thought of as the generalization of Fourier modes (circular harmonics) to the sphere. Any real-valued function on the sphere $f : \mathbb{S}^2 \to \mathbb{R}$ can be Fourier Transformed to this SH basis:

$$\mathscr{F}(f)(l,m) = \hat{f}_m^l = \int_{\mathbb{S}_2} f(\mathbf{n}) Y_m^l(\mathbf{n}) \mathrm{d}\mathbf{n} \tag{2.60}$$

$$f(\mathbf{n}) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \hat{f}_m^l Y_m^l(\mathbf{n}) \quad \text{(inverse transform)} \tag{2.61}$$

where $\mathbf{n} = (\theta, \phi) \in \mathbb{S}$ represents a generic direction or point on the sphere. Here the coefficients $\hat{f}^l$ are not real values but are $(2l+1)$-dimensional vectors (with components $\hat{f}_m^l$). Each coefficient $\hat{f}^l$ transforms according to a Wigner-D matrix D^l: the SH embedding is thus equivariant. Note that to make a representation finite-dimensional, we need to choose a high-frequency cutoff for the rotational order, $l = l_{max}$.

Coming to implementation, a feature is just a concatenation of different l-vectors: scalars ($l = 0$), 3-D vectors ($l = 1$), 5-D vectors ($l = 2$) and so on. Multiple pieces with the same l are also allowed, we address this multiplicity by referring to channels. For example we can have two $l = 0$ channels, a single $l = 1$ channel and a single $l = 2$ channel:

$$\mathbf{h}(\mathbf{x}) = \left(h_{c=0}^{(l=0)}(\mathbf{x}), h_{c=1}^{(l=0)}(\mathbf{x}), \mathbf{h}_{c=0}^{(l=1)}(\mathbf{x}), \mathbf{h}_{c=0}^{(l=2)}(\mathbf{x})\right) \tag{2.62}$$

where $\mathbf{h} : \mathbb{R}^3 \to \mathbb{R}^k$ with $k = \sum_l n_c^{(l)} \cdot (2l+1) = 2 \cdot 1 + 3 + 5 = 10$ with $n_c^{(l)}$ number of channels of type l. Rotation of these features is straightforward:

$$D(r)\mathbf{h} = \begin{bmatrix} D^0 & & & \\ & D^0 & & \\ & & D^1 & \\ & & & D^2 \end{bmatrix} \begin{bmatrix} h_{c=0}^{(l=0)} \\ h_{c=1}^{(l=0)} \\ \mathbf{h}_{c=0}^{(l=1)} \\ \mathbf{h}_{c=0}^{(l=2)} \end{bmatrix} \tag{2.63}$$

The representation matrix is block-diagonal thanks to $SO(3)$ being decomposed in a direct sum, and scalars ($l = 0$) being invariant with respect to rotation ($D^{(0)} = 1$).

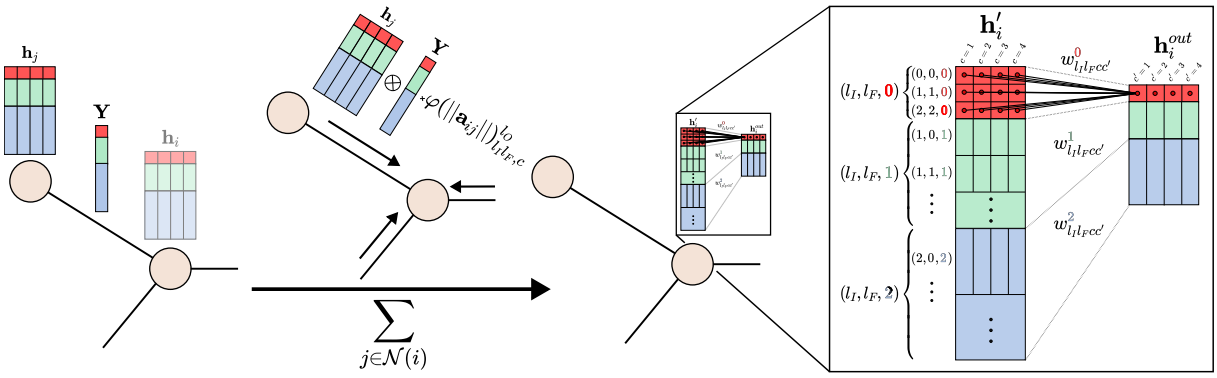


Figure 2.6: **Overview of the convolution layer, summarizing Eqs. (2.64,2.65).** For each neighboring node, the node and edge features are combined (with C-G product) and multiplied by the learned radial filter φ. Before performing this operation, the one-hot encoded particle type is concatenated to $\mathbf{h_i}$ by adding 2 $l = 0$ channels (not shown, for simplicity). Because multiple triplets come out of the C-G product, we obtain a much larger representation (left part of inset). This intermediate representation is narrowed down using a linear layer (one for each l_O and each channel).

Convolution Layer We can now adapt the definition of steerable convolutions (2.3.3) to the GNN set-up. Here we introduce multiple channels indexed by c and we split the full operation in two steps: the *convolution* and the *self-interaction*. The integration over the space variable $\tilde{\mathbf{x}} \in \mathbb{R}^3$ turns into a sum over neighbors due to point-wise nature of the feature field and the convolution operations reads as:

$$\mathbf{h}'^{l_O}_{i,c,l_I l_F} = \sum_{j \in \mathcal{N}(i)} \varphi(\|\mathbf{a}_{ij}\|)^{l_I,l_O}_{l_F,c} \mathbf{Y}^{l_F}(\hat{\mathbf{a}}_{ij}) \otimes^{l_I,l_O}_{l_F} \mathbf{h}^{l_I}_{j,c} \tag{2.64}$$

The radial filters $\varphi^{l_I,l_O}_{l_F,c}$ are implemented as Multi-Layer Perceptrons (MLPs, to be learned) that share some weights among triplets l_O, l_I, l_F and channels c. This operation is depicted in Figure 2.6 (left part). The updated feature $\mathbf{h}'^{l_O}_{i,c,l_I l_F}$ at node i is indexed by the channels (which are in one-to-one correspondence with the input ones) and by the triplet of l's. At this stage, operations are performed channel-wise, but $\mathbf{h}'_{i,c}$ is a concatenation of all possible triplets, and as multiple combinations of l_I, l_F can contribute to a given l_O, it is larger than the original feature $\mathbf{h}_{i,c}$. For $l_{max} = 3$, there are 34 different triplets (instead of just 4 different values of l).

To go back to the original representation, we mix together the triplets that share the same output l_O, with a linear layer. However, to let the various channels interact, we also perform

channel mixing with a linear layer. As linear layers combine linearly, this can be expressed as a single linear layer (right part of Figure 2.6) that is often referred to as the self-interaction layer:

$$\mathbf{h}_{i,c}^{\text{out},l_O} = \sum_{l_I l_F, c'} w_{l_I l_F, cc'}^{l_O} \mathbf{h}_{i,c',l_I l_F}^{\prime l_O} \tag{2.65}$$

where c' is the input channel's index and c is the output one. Note that all operations are now performed node-wise, and independently for each l_O. This operation fulfills equivariance because only features with the same l_O are combined together, with weights that do not depend on m (all elements inside a vector are multiplied by the same factor). At this point we are back to our expected node feature shape, and the convolution layer can be repeated (up to a few technical details like using Batch-Norm and adding the previous layer's representation to the newly computed one, see next section).

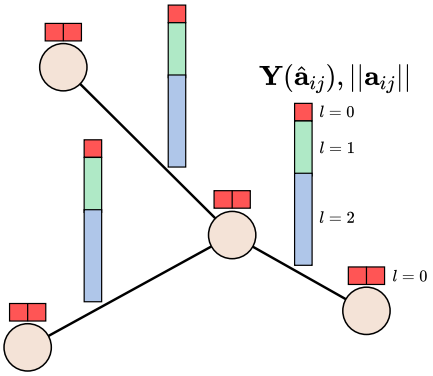


Figure 2.7: **Input Graph with its input features.** Node features are the one-hot encoded particle types (invariant features, $l = 0$), and edge attributes $\mathbf{a_{ij}}$ are split: the direction is embedded in Spherical Harmonics $Y(\hat{\mathbf{a}}_{\mathbf{ij}})$ and the norm is retained separately.

2.4 SE(3)-equivariant GNN for learning glassy liquids representations

In this section, we present the methods and results adopted in our work on rotation-equivariant graph neural networks for learning glassy liquid representations [PCL24].

2.4.1 Dataset and Task

To probe the ability of our model to predict mobility, we adopt the dataset built by Bapst *et al.* in [BKGB+20]. It is obtained from molecular dynamics simulations of an 80:20 Kob-Andersen mixture of $N = 4096$ particles in a three-dimensional box with periodic boundary conditions, at number densities of $\rho \simeq 1.2$. Four state points (temperatures) are analyzed: $T = 0.44, 0.47, 0.50, 0.56$. For each point, 800 independent configurations $\{\mathbf{x}_i\}_{i=1...N}$ are available, *i.e.* 800 samples (each sample represents N particles' positions).

The quantity to predict (Ground Truth label) is the individual mobility of each particle, measured as the dynamical propensity [BJ07, BBB+07]: for each initial configuration, 30 micro-canonical simulations are run independently, each with initial velocities independently sampled from the Maxwell-Boltzmann distribution. The propensity of particle i over a timescale τ is then defined as the average displacement over the 30 runs (iso-configurational ensemble average). Propensity is available at $n_{times} = 10$ different timescales that span the log scale, *a priori* resulting in n_{times} different tasks. For some experiments we also use another similar dataset as provided by Shiba *et al.* [SHSS23], which models the same glass-former yet differs from that of Bapst *et al.* on a couple of points, that we detail in section 2.4.3. Note that the finite number of independent runs (here, 30) in the iso-configurational ensemble induces some noise in the estimation of the propensity. This uncertainty in our ground truth induces an upper bound on the theoretically achievable accuracy of any prediction method. This bound has been computed in [JAB+23]; we do not report it here, as we are far enough from it, in order to avoid obscuring the figures.

For each sample to be processed through the GNN, the input graph is built by taking particles

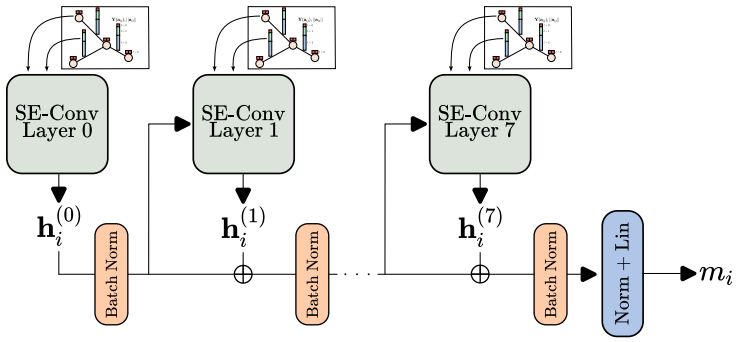


Figure 2.8: **Overall Architecture.** Top: node and edge features are fed to each convolution layer. Each SE-convolution layer $L = 0, \ldots, 7$ refines the output $\mathbf{h}_i^{(L)}$. Arrows connecting the embedded graph features to each convolution block show that the initial information (one-hot particle types and relative positions in SH and radial basis) is fed to each layer.

as nodes and connecting them when the inter-atomic distance between positions $\mathbf{x}_i$ and $\mathbf{x}_j$ is less than $d_c = 2$ (in atomic potential units). The node features encode the particle type, here A or B ("Node features" is machine learning vocabulary for "set of values associated to the node", and similarly for edge features). We use one-hot encoding, such that node features consist of $n_{type} = 2$ boolean variables. This generalizes trivially to mixtures with $n_{type} > 2$. Optionally, we also include the value of the potential energy of particle i as node feature, which brings their number to 3 (2 boolean and a real). The edges are directed, and edge (i, j) has for feature $\mathbf{a}_{ij} = (\mathbf{x}_j - \mathbf{x}_i)$, *i.e.* it stores the relative position of the particles (nodes) it connects. We show a sketch of our input graph with its node and edge features in Figure 2.7.

The task is then the node-wise regression of the particle's propensity $m_i \in \mathbb{R}$ (node label). Notably, here we simultaneously regress both particle types, meaning that all nodes contribute to the computation of the loss function. We also introduce a new task, referred to as *multi-variate regression*, in which the n_{times} timescales are regressed at once, as opposed to the usual *uni-variate* approach.

2.4.2 Network

Our network is composed of embedding blocks for nodes and edges features followed by a series of SE(3)-equivariant convolutional layers interspersed with batch normalization and connected in a Res-Net fashion [HZRS15], and one output block (decoder), as shown in Figure 2.8. Here we provide a few insights on some key parts that are specific to SE(3)-equivariant networks.

The architecture choice is found empirically to be the most stable at train time. It is built and trained in the framework of PyTorch Geometric [FL19] which handles all the generic graph operations. All the SE(3)-related operations (SH embedding, C-G tensor product, equivariant batch-normalization) are integrated in this framework thanks to the *e3nn* library [GSM+22].

Radial MLP The radial MLPs are the only part of the network with non-linearities. They implement the radial dependence of convolution filters $\varphi_{l_F,c}^{l_I,l_O}(\|\mathbf{a}_{ij}\|)$, thus they take as input the norms of relative node positions. Before being fed to the MLP, each norm $\|a_{ij}\|$ is expanded

from a real value to an array through an embedding. Here we use a Bessel basis embedding:

$$B_n(r) = \sqrt{\frac{2}{r_c}} \frac{\sin\left(n\pi \frac{r}{r_c}\right)}{r} \tag{2.66}$$

where n is the number of roots of each basis function: $n = 1, 2, \ldots, N_b$ and we use $N_b = 10$, $r_c = d_c = 2$. Other embeddings could be for instance a Gaussian basis (with cutoff), which would act as a kind of smooth one-hot encoding of the value r. In practice, the Bessel basis (which is orthogonal) has better generalization properties.

The embedded input is processed through an MLP with layers sizes $(N_b, 16, n_{comb})$ and ReLu non linearities. The output size n_{comb} is the number of possible triplets (combinations), times the number of channels. We also use BatchNorm (BN) [IS15] and Dropout (with rate $p = 0.3$) in this MLP to stabilize training and reduce overfitting. In summary, for each combination of triplets and channels (l_O, l_I, l_F, c), we have a real output

$$\varphi^{l_I,l_O}_{l_F,c} = \sigma(W_{n_{comb},16} Dropout(BN(\sigma(W_{16,N_b} B(||\mathbf{a}_{ij}||)))))) \tag{2.67}$$

where the W's are weight matrices and $\sigma(z) = max(0, z)$. There are also bias parameters, which are not displayed here. Note that up to the layer of 16 neurons, the MLP is the same for all triplets and channels, only the last linear layer introduces different weights for each combination.

Batch Normalization As often in Neural Networks, we sometimes need to perform Batch Normalization to avoid the neuron's activation to take overly large values. However, using a usual batch normalization layer [IS15] separately on each entry of the hidden representations $\mathbf{h}$ would kill the equivariance property. Thus a modified version is implemented and applied to node features [WGW+18, GSM+22]. The $l = 0$ features are invariant and can be processed as usual:

$$h^0_{BN} = \frac{h^0 - \bar{h}}{\sigma}\beta + \gamma \tag{2.68}$$

where $\bar{h} = \langle h^0 \rangle$ and $\sigma^2 = \langle {h^0}^2 \rangle - \langle h^0 \rangle^2$ with $\langle \cdot \rangle$ batch average computed with 0.5 momentum (keeping memory of previous batches) and β, γ are learned parameters. For each piece of feature with $l \geq 0$, only the norm can be modified:

$$\mathbf{h}^l_{BN} = \mathbf{h}^l \frac{||\mathbf{h}^l||}{\sigma^l}\beta^l \tag{2.69}$$

where $\sigma^l = \sqrt{\langle ||\mathbf{h}^l||^2 \rangle}/\sqrt{2l+1}$ and β^l are learnable parameters. In Figure 2.8 we show where this Batch Norm is used.

Decoder After the last convolution layer, the ultimate output block performs two node-wise operations to decode the last layer's output into a mobility prediction. First it computes SE(3)-invariant features from the hidden representation $\mathbf{h}^{(L_{max})}$: for each channel $c = 1, \ldots, 8$, the norm of each directional ($l \geq 1$) feature is computed: $||\mathbf{h}^l||_2 = \sqrt{\sum_m {h^l_m}^2}$, and all these norms are concatenated together with the $l = 0$ features (already invariant). Thus, we obtain an invariant

representation of exactly $l_{max}+1$ (l values) $\times 8$ (channels) $= 32$ (components), which we denote $|\mathbf{h}^{(L_{max})}|$ for simplicity, despite the fact that the $l=0$ components can be negative. The second operation is to feed this representation into a decoder, which we chose to be a linear layer, it outputs one real value, which is the predicted mobility for a given timescale and given particle type. For instance, at the timescale τ_α and for particles of type A, the model writes:

$$y_{A,\tau_\alpha} = \mathbf{w}_{A,\tau_\alpha} |\mathbf{h}^{(L_{max})}(\{\mathbf{x}\})|, \tag{2.70}$$

where y_{A,τ_α} is the mobility label and $\mathbf{w}_{A,\tau_\alpha}$ is a set of weights to be regressed (32 real values). In the multi-variate setup we regress mobilities at all timescales at once, using one linear decoder (set of weights $\mathbf{w}$) per timescale and per particle type (20 different decoders for the Bapst dataset).

Non-linearities We note that all the layers act linearly on the node features. The only non-linearities of the network are hidden in the implementation of the radial part of the filters φ (MLPs). This limited scope of non-linearities is unusual, and is needed to preserve equivariance (as pointed out above when we describe Batch Norm). We have explored other forms of non linearities, like Gate-activation, without observing significant improvement.

Connection with expert features We emphasize that the first layer of our network has a clear interpretation, connecting with the widely-used expert features introduced in Sec. 2.2.1. In the first convolution, the input node features $h_{i,c} = \delta_{t_i,c}$ consist of two $l=0$ channels, corresponding to the one-hot encoding of the particle type. Since $l_I = 0$, the only non-zero C-G triplets occur when $l_O = l_F = 0, 1, 2, 3$, and for these triplets, $[CG]_{l_F}^{l_O=l_F, l_I=0} = \mathbb{I}$. Thus, the graph steerable convolution defined in Eq. 2.64 simplifies to:

$$\mathbf{h}_{i,c}^{'l_F} = \sum_{j \in \mathcal{N}(i)} \varphi(\|\mathbf{a}_{ij}\|)_{l_F,c} \delta_{t_j,c} \mathbf{Y}^{l_F}(\hat{\mathbf{a}}_{ij}) \tag{2.71}$$

For the channel corresponding to particle type A, i.e., $c = A$, and by fixing the radial function to be constant, $\varphi(\|\mathbf{a}_{ij}\|)_{l_F,c} = 1$, we obtain:

$$\mathbf{h}_{i,A}^{'l_F} = \sum_{j \in \mathcal{N}(i), t_j = A} \mathbf{Y}^{l_F}(\hat{\mathbf{a}}_{ij}) \tag{2.72}$$

This is in the same form as the BOP coefficients adopted in [BMAM+20, CJP22]. This operation projects all neighbors of the central particle onto the unit sphere within a cutoff radius. However, as discussed in the cited works, these descriptors are limited as they lose information about different neighbor shells by squashing the neighborhood onto the unit sphere. To address this, descriptors like SOAP [CJP22] and RBO [BSF21] were introduced, which incorporate radial information. In these approaches, radial dependence is included by introducing a Gaussian radial basis to separate neighbors into different shells. For each shell, a quantity analogous to Eq. 2.72 is computed. In [BSF21], each shell is treated separately, while in [CJP22], multiple shells interact. Specifically, the SOAP coefficients from [CJP22] take the form:

$$c_{nlm}(i) = \sum_{j \in \mathcal{N}(i)} g_n(\|\mathbf{r}_{ij}\|) Y_{lm}(\hat{\mathbf{r}}_{ij}), \tag{2.73}$$

where g_n is the n-th radial basis function corresponding to distance r from the central particle. The interaction between shells occurs when computing the product $c^*_{nlm}(i)c_{n'lm}(i)$.

Our network can similarly retrieve the $c_{nlm}(i)$ coefficients by learning the radial function or replacing it with a radial basis function (see the Appendix of [PCL24] for a detailed comparison with [BSF21]). However, when it comes to combining different shells the approach of our work is quite different. By expanding the function φ, which is implemented as an MLP acting on a radial basis expansion of the vector $\mathbf{a}_{ij}$, we can rewrite the equation as: (we assume simplest MLP with only one linear layer)

$$\mathbf{h}^{'l_F}_{i,c} = \sum_{j\in\mathcal{N}(i)} \sum_n w_{n,l_F,c} B(||\mathbf{a}_{ij}||)_n \delta_{t_j,c} \mathbf{Y}^{l_F}(\hat{\mathbf{a}}_{ij}). \tag{2.74}$$

In the notation of [CJP22], this corresponds to the following operation:

$$\mathbf{h}^{'l_F,m_F}_{i} = \sum_n w_{n,l_F,c} c_{nl_Fm_F}(i), \tag{2.75}$$

which amounts to summing different shells with learned weights for each. Additionally, our approach introduces non-linear functions for the radial part, multiple channels, and channel mixing. However, the main distinction lies in how we treat equivariant features: while [BSF21, CJP22] compute invariant descriptors by taking the norms of l-th order features, our network maintains equivariant features throughout, only computing the invariant features at the final layer.

This distinction significantly enhances our model's ability to predict mobility. For a detailed discussion on performance improvements and an ablation study, refer to Sec. 2.4.3.

Number of parameters This counting refers to the version of our network with 8 channels and no E_{pot} in input. In total, the MLPs of our network (across all layers) account for a number of 35 664 learnable parameters: in each layer $L > 0$ we have one radial MLP of size $(10, 16, 284)$ with 5036 parameters, for the layer $L = 0$ the MLP is of size $(10, 16, 3 \times 4)$ with 412 parameters. The other main source of learnable parameters in the Network is the part of mixing the channels (right part of fig. 2.6), which accounts for 16000 learnable parameters: 2272 for each $L > 0$ layer and $12 \times 8 = 96$ for the $L = 0$ layer. The total number of parameters to build the representation is thus $35664 + 16000 = 51664$. One has to add to this number the parameters of the 20 decoders (10 timescales for A and B particles). The final number is then: $51664 + 32 \times 20 = 52304$. When single variate (single time scale) regression is performed (as in the other GNNs works we compare with), the number of channels is reduced to 4 and the total number of parameters amounts to 23210.

2.4.3 Experiments, Results

Here we report on the performance of our architecture, discuss the role of the task, input choices and architecture choices, and compare with recent works that tackle the same problem.

Experimental Setup and Training Strategy To increase our model's robustness, we simultaneously predict the mobility both for A and B particles, instead of focusing only on the

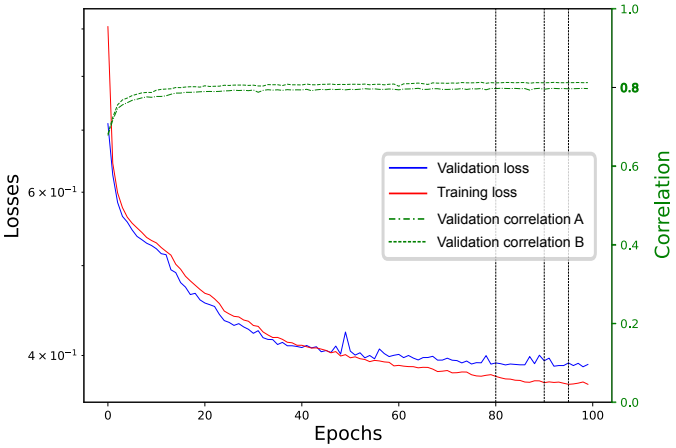


Figure 2.9: **Loss and ρ vs epoch.** Training of multi-time model performed at $T = 0.44$. The loss curves (full lines) correspond to the total loss of the multi-variate regression setting: sum over all the timescales of the mean squared error per timescale. The correlation curves show the Pearson correlation coefficient between predicted mobility and the ground-truth for a single timescale $\tau = \tau_\alpha$. Vertical dashed lines locate the epochs at which the learning rate is divided by 2.

A's. The accuracy turns out to be similar for the two types. We show results only for one type, A, which is the reference most other works are also using. As in the works we compare with, we use the Pearson correlation coefficient as performance metric, which is invariant under shift and scale of the test labels distribution. The network architecture and hyper-parameter choices were optimized for a single task ($T = 0.44$ and $\tau = \tau_\alpha$ for uni-variate and $T = 0.44$ for multi-variate), using only the train and validation sets. The resulting choices were applied straightforwardly to other tasks, thus preventing over-tuning of the hyper-parameters. The number of convolution layers is 8, thus the last representation is indexed $L = L_{max} = 7$ (representation $\mathbf{h}_i^{(L=0)}$ at $L = 0$ is the input, before any convolution). At each layer $L > 0$ the internal or hidden representation $\mathbf{h}_i^{(L)}$ for particle i at layer (L) has a maximum rotational order $l_{max} = 3$ and a number $n_c^{(l)} = n_c^{(0)} = n_c^{(1)} = n_c^{(2)} = n_c^{(3)}$ of channels, $n_c = 4$ for uni-variate and $n_c = 8$ for multi-variate. These choices arise from striking a balance between over- and under-fitting, under our compute-time and memory budget constraints.

Note that we perform a different train-test split with respect to [BKGB+20], which does not explicitly use a test set. Here, for each state point, 400 configurations are used for training, 320 for validation and 80 for the final test.

In Figure 2.9 we display one learning curve (as function of iterations, epochs). Each epoch is a sweep over the entire 400 samples dataset (each sample represents $N = 4096$ atoms). For training, we use the Adam optimizer with initial learning rate $\gamma = 10^{-3}$, moments $\beta_1 = 0.99, \beta_2 = 0.999$ and weight decay $\lambda = 10^{-7}$. We also add a learning rate scheduler that divides γ by 2 at several epochs as shown by the vertical dashed lines in Figure 2.9. Most of the results presented here are obtained with a number of epochs $n_{epochs} = 100$, this choice results from several tests and strikes the balance between accuracy and training time. As it can be seen in Figure 2.9, each training stops before any serious overfit kicks in.

Uni-variate or Multi-variate In Figure 2.10 we compare the performances of various choices for our model, in particular uni-variate and multi-variate approach (red triangle and red diamonds). We see that we get almost the same prediction accuracy by training only one model instead of ten models, provided we increase the number of parameters for that single model: we double the number of channels in the multi-variate case, from 4 to 8, thus going from ~ 25000

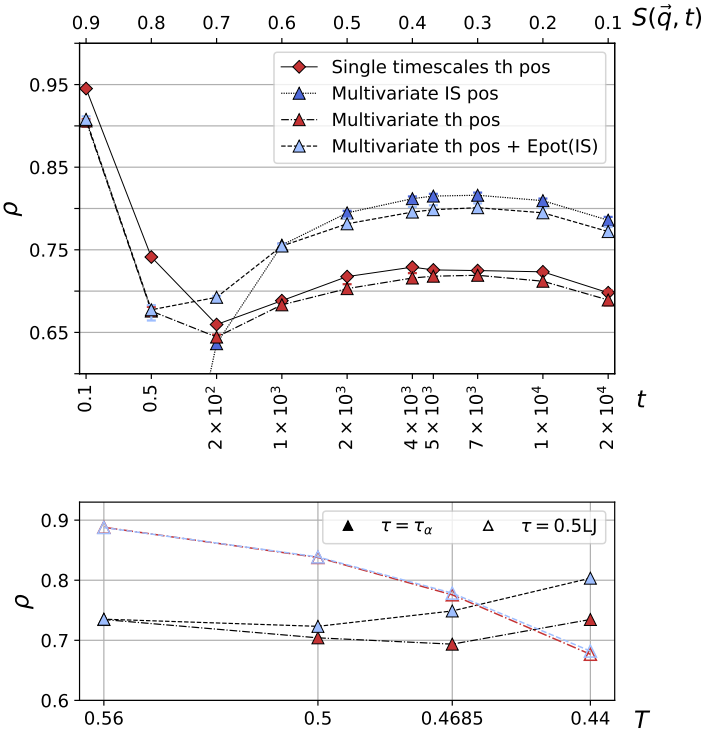


Figure 2.10: **Multi-variate vs univariate and influence of inputs.** (top) Correlation ρ between the true and the predicted propensity for A particles at temperature $T = 0.44$ as function of timescale. Marker shapes distinguish multi-variate and uni-variate approaches. Colors picture the input type: red for thermal positions ($\{\mathbf{x}_i^{th}\}$), blue for quenched (Inherent Structures, IS) positions ($\{\mathbf{x}_i^{IS}\}$) and light-blue for combined ($\{\mathbf{x}_i^{th}\} + E_{pot}^{IS}$). Error-bars represent the best and the worst ρ for ten identical models trained independently with different random seed initialisation, and are comparable with marker's sizes. (bottom) Correlation ρ as function of training (and testing) temperature. Two timescales are shown: $\tau = \tau_\alpha$ (full markers) and $\tau = 0.5\tau_{LJ}$ (empty markers). Color code and marker code identical to that of the top plot. The multi-variate, thermal positions + $E_{pot}(IS)$ choice is a good compromise to maintain high performance across timescales.

to ~ 50000 parameters. Actually, also keeping the number of parameters constant yields competitive results (not shown). In any case, we observe that the multi-variate choice slightly improves the robustness of our representation: it generalizes better to other temperatures. Beyond performance considerations, it is very advantageous when considering generalization to other temperatures, since all timescales are encompassed in the same representation $|\mathbf{h}^{(L_{max})}|$. In this sense, our network is about an order of magnitude less parameter-hungry that other models, where each of the 10 timescales and each particle type need a dedicated network.

Role of Inherent Structures It has been observed several times that pre-processing the input positions by quenching them to their corresponding Inherent Structures (IS) helps most Machine Learning models in predicting long-time mobility measures [SCS+16, JBB23, ASF23]. Such a quench is performed using the FIRE algorithm: temperature is set to 0 (velocities set to 0), and positions adjust gradually so as to converge to a local minimum of the potential energy, typically close to the original configuration. This can be seen as a mere pre-processing step (for which the knowledge of the interaction potentials is needed) or as a new task, *i.e.* predicting the propensities $\{m_i\}$ from the quenched positions $\{\mathbf{x}_i^{IS}\}$. We note that the quench, while intuitively cleaning some noise related to thermal motion, destroys information too: one cannot recover the thermal positions from the quenched ones.

We observe that for our network, this new task is harder at short timescales, while it's easier at long timescales (in Figure 2.10, compare the red diamonds and the dark blue downward-pointing triangles). We interpret this result by noting that the quench destroyed the information about

	$t = 0.5LJ$		$t = 1\tau_\alpha$	
	$\{\mathbf{x}_i^{th}\}$	$\{\mathbf{x}_i^{IS}\}$	$\{\mathbf{x}_i^{th}\}$	$\{\mathbf{x}_i^{IS}\}$
no E_{pot}	$0.676^{+0.005}_{-0.006}$	$0.274^{+0.001}_{-0.002}$	$0.718^{+0.007}_{-0.003}$	$0.815^{+0.003}_{-0.003}$
E_{pot}^{th}	$0.678^{+0.006}_{-0.014}$	$0.334^{+0.002}_{-0.001}$	$0.728^{+0.005}_{-0.003}$	$0.816^{+0.003}_{-0.001}$
E_{pot}^{IS}	$0.677^{+0.005}_{-0.013}$	$0.273^{+0.001}_{-0.002}$	$0.798^{+0.005}_{-0.002}$	$0.822^{+0.003}_{-0.001}$

Table 2.1: **Influence of IS at low temperature.** For each combination of inputs, a multi-time model is trained at temperature $T = 0.44$. We repeat the training 10 times with variable parameter initialization and report the test set correlation coefficient (median, best and worst values).

the relative location of each particle within its cage, thus making it much harder to predict short-time displacements. Our experiment and its interpretation explain why some models, based on quenched positions alone, have very low performance at short timescales [JBB23]. Their low performance should not be attributed to the machine learning models themselves, but rather to their input data. About mobility at long times, there it is not much of a surprise that quenched positions reveal an underlying slowly-evolving pattern in the structure and thus help at prediction (although in principle all the information was contained in the original thermal positions).

Ideally, one would like to combine both the complete information from thermal positions and the de-noised information from the quenched positions. For GNNs, this could be done by building the graph from either the thermal or quenched relative positions, but using as edge features a concatenation of both. However this would be quite costly in terms of memory and would increase the number of parameters needlessly. Instead, inspired by the findings of [JBB23], we compute the local potential energy (in either the thermal or the IS positions) for each particle $E_{pot,i} = \sum_{j \neq i} V_{LJ}(\mathbf{x}_i, \mathbf{x}_j)$ and use it as a new scalar ($l = 0$) input node feature. This can be seen as a compressed version of the positional information. Note that the first layer remains very interpretable: this new channel represents the field of potential energies surrounding a given particle, expressed in the spherical harmonics basis. In Table 2.1 we compare performances obtained for all combinations of input positions (thermal or quenched), with all possible E_{pot} inputs (none, thermal or quenched), resulting in 6 combinations, that we study at two timescales: $0.5\tau_{LJ}$ and τ_α. We summarize the key results from this table:

- Adding the information about E_{pot}^{IS} to $\{\mathbf{x}_i^{IS}\}$ is irrelevant. Indeed we observed that we could easily regress E_{pot}^{IS} from a network with $\{\mathbf{x}_i^{IS}\}$ input with very high precision ($\rho \approx 0.9$).
- Similarly for thermal positions and thermal potential: adding E_{pot}^{th} to $\{\mathbf{x}_i^{th}\}$ is basically useless, the increase from $\rho = 0.718$ to 0.728 is barely statistically significant.
- Adding E_{pot}^{th} to $\{\mathbf{x}_i^{IS}\}$ helps only at short timescales (from $\rho = 0.27$ to 0.33) and it's not sufficient to fill the gap with thermal positions.
- Adding E_{pot}^{IS} to $\{\mathbf{x}_i^{th}\}$ helps, but at long timescales only (from $\rho = 0.72$ to 0.80)
- For predicting short times, thermal positions work much better than quenched ones: 1st column shows consistently larger performance than the 2nd one, by up to 0.4 more in correlation.

- For predicting long times, quenched positions work better than thermal ones: 4th column shows consistently larger performance than the 3rd one, by up to 0.1 more in correlation.
- A good compromise for maintaining performance at all timescales is to combine E_{pot}^{IS} to $\{\mathbf{x}_i^{th}\}$.

In the table we focus on two timescales for clarity, and in Figure 2.10 (top) we report results for 3 out of the 6 combinations but at all times. In Figure 2.10 (bottom) we study the effect of adding E_{pot}^{IS} to the thermal positions (red to blue symbols) as a function of temperature, for two timescales. We verify that for the long timescale (full symbols) the addition of E_{pot}^{IS} helps especially for the lower temperatures, where the potential energy landscape is expected to be more relevant, while for the short timescale (open symbols) there's no improvement at all, at any temperature.

We can compare these observations with the findings of Alkemade *et al.* [ASF23]. They identify three physical quantities, each being relevant in a given time range:

1. In the ballistic regime, the forces $\mathbf{F}_i = -\nabla_{\mathbf{x}_i} E_{pot,i}$ are most relevant
2. In the early caging time, the distance between the thermal position and the IS one Δr^{IS} is most relevant
3. At later times, the quenched configurations are most relevant

For the ballistic regime, our results perfectly match theirs: our model is likely to be aware of information equivalent to the forces, since it's able to regress the local potential energy with very high accuracy ($\rho \approx 0.9$). This explains our good performances in the very early regime (see also Figure 2.11). For the early caging regime, we tried to introduce Δr^{IS} as a further $l = 0$ node feature but were not able to see any significant improvement in the caging regime. This may be due to improper encoding of this information, or to a deeper shortcoming of our architecture, or also to the datasets being slightly different (see paragraph *Shiba vs Bapst*). For the long times, our performances are indeed high thanks to the use of E_{pot}^{IS}: they are slightly higher if we use $\{\mathbf{x}_i^{IS}\}$ (see Table 2.1 or Figure 2.10 (top)).

Comparison with recent works Often, comparison with previous works can not be done rigorously, for two reasons: use of different datasets and different input data. As mentioned in the previous section and already pointed out in [ASF23] the two main datasets [BKGB$^+$20, SHSS23] differ in many features, although being built from MD simulations of the same system (3D Kob-Andersen mixture). A detailed comparison at fixed input dataset is presented in a Roadmap paper [JAB$^+$23]. A further difference is introduced by the choice of input data. For instance we have shown that the introduction of Inherent Structures helps, especially for low temperatures and long timescales. Thus better performances for works that rely on IS do not directly imply that the machine learning architecture is better (or vice versa for works that are limited to thermal inputs).

Despite these limitations, in Figure 2.11 we provide a qualitative comparison of methods by considering each as a whole, regardless of the details of dataset and input choice. Thus we compare our model trained on thermal positions + $E_{pot}(IS)$ at temperature $T = 0.44$, with the recent works in the field presented in Sec 2.2.3 [BSF21, ASF23, JBB23, BKGB$^+$20, SHSS23].

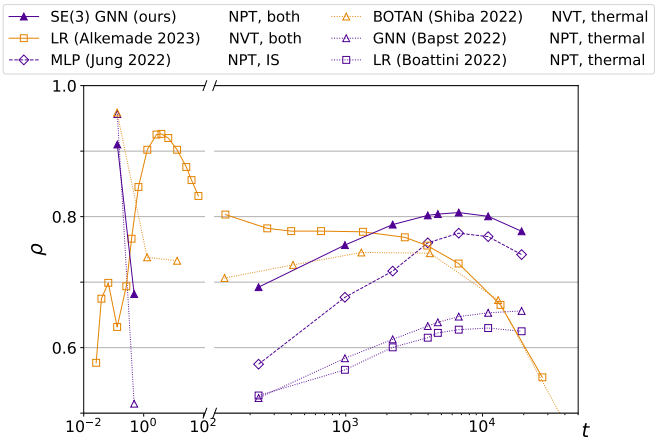


Figure 2.11: **Comparison with recent works.** Correlation ρ between true and predicted propensity for A particles at temperature $T = 0.44$, as a function of timescale, for several recent works. Color indicates dataset choice: dark purple for Bapst' (NPT equilibration), orange for simulations using NVT equilibration. Line-styles indicate input choices: thermal data only (dotted lines), IS data (dashed lines) or a combination of both (solid line). Markers describe the type of model: upper triangles refer to GNNs, diamonds for MLPs and squares for Linear Regression. The grey shaded area locates τ_α which is slightly different for the two datasets. Note how curves computed for a given dataset barely cross each other, indicating rather consistent ranking between models.

Our proposed approach outperforms all previous methods for timescales approaching the structural relaxation time (τ_α), while demonstrating competitive results in other regimes. Notably, our model achieves comparable performance to other GNN approaches on short timescales (ballistic motion), despite being the first to regress all timescales simultaneously. For the early caging regime, we do not perform as well as Alkemade *et al.* [ASF23] although it is important to note that they incorporate early-times related information as an input feature. We overperform Shiba *et al.* [SHSS23] only when using the quenched input. We do not know about their performances when using the quenched input too.

Since the first version of this work (preprint of Nov. 2022), we tried to include these recent works' ideas to improve performance. Our use of E_{pot}, inspired by [JBB23], was indeed successful. However, when we tried to mimick [SHSS23] by regressing the edge relative elongation as an additional (edge) target label, or when we tried to reproduce the results of [ASF23], using as input node feature the distance to the local cage center (estimated as the quenched position), or when we introduced equivariant attention schemes (inspired by [JTL22] but technically as in [LS22, HLLZ+21] or [FWFW20]), our attempts did not yield any significant improvement (nor deteriorated the performance).

To compare Machine Learning architectures in a fair way, one should work at fixed task (fixed dataset and input data). We now respect this constraint to obtain two precise results, that we deem significant.
Firstly, going back to using only the thermal positions as input, we perform an ablation study on the choice of l_{max}, to compare fairly with Bapst *et al.*, and notice that: (i) restricted to $l_{max} = 0$, we reach the same accuracy, (ii) increasing l_{max} notably improves results, especially up to $l_{max} = 2$. We conclude that the equivariant nature of a network can be key to its performance, compared to previous GNN approaches. Numerical proof is provided in the *Ablation Studies* paragraph.
Secondly, using the same kind of (invariant) inputs as non-GNN methods [BSF21, ASF23, JBB23], *i.e.* thermal positions combined with $E_{pot}(IS)$, in addition to the (equivariant) po-

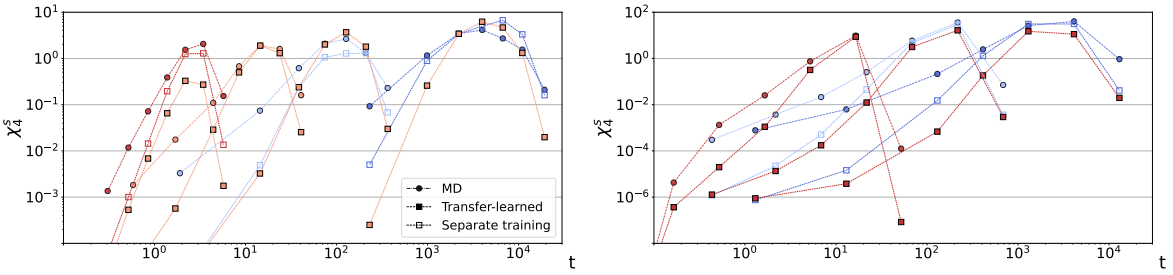


Figure 2.12: **Fluctuations of the Self-overlap function.** Time evolution of the fluctuations as measured by $\chi_4^s(t)$. Left: On Bapst's dataset; Right: Shiba's dataset. MD is short for Molecular Dynamics and refers to the ground truth. "Separate training" indicates a new model was trained at each temperature (but dealt with all timescales at once), while "transfer-learned" refers to sec. 2.4.4: we apply a single model trained at a given temperature to all other temperatures (left: $T_{train} = 0.50$, right: $T_{train} = 0.56$). The color code represents training temperatures, ranging from red (hottest) to blue (coolest) for each dataset.

sitional inputs, we study the impact of the network's depth. We noticed already in Figure 2.11 that we perform better than those methods, at most timescales. Here we want to stress that the network's depth plays a crucial role (more so that the rotational order l_{max}): varying the number of convolution layers from $L_{max} = 1$ to $L_{max} = 7$, we noticed that performance does not even saturate. We conclude that although using invariant features (and ideally, equivariant ones) is helpful, the combinatorial power of deep learning architectures is also key to performance. Numerical proof is provided in the *Ablation Studies* paragraph.

A side result of these ablation studies is that the short timescales seem to be the ones that benefit the most from increased l_{max}, while they also benefit from increased depth (L_{max}), up to saturation at $L \geq 4$. We conjecture that directional features are key to computing instantaneous forces, itself a key element for predicting short-time dynamics.

Spatio-temporal correlations The particle-wise correlation coefficient between ground truth mobility and predicted one is not everything, it's good to also measure whether the statistical properties of our predicted mobility match those of the true one.
Defining $c_i(t) = \tanh\left(20(m_i(t) - 0.44) + 1\right)/2$ a pseudo-binarized mobility measure, $Q^s(t) = \frac{1}{N_A}\sum_{i\epsilon A} c_i(t)$ its sample average (also called self-overlap function), one defines a four-point correlation function $\chi_4^s(t) = N_A\left[\left\langle Q^s(t)^2\right\rangle - \left\langle Q^s(t)\right\rangle^2\right]$, the fluctuations of the Self-overlap function, that we report in Figure 2.12 (we use the same specifications as in [JBB23]). This measure of the sample-to-sample fluctuations of mobility is often interpreted as a volume of correlation (as it can be re-written as the integral of a correlation function). Our estimated χ_4^s ("separate training") is generally smaller than the ground-truth (MD) but tracks variations over time fairly well and much better so than the initial GNN of [BKGB+20]. Furthermore it is comparable to the performance of [JBB23], which however incorporates information about fluctuations in theirs model's loss. One may notice that the amplitude of fluctuations is smaller in the first dataset (Baspt's): this is due to the peculiar sampling choice, in which samples at a given "timescale" are actually taken at different times but equal value of the self-intermediate scattering function $F_k(t)$, a choice which by definition reduces the variance between samples.

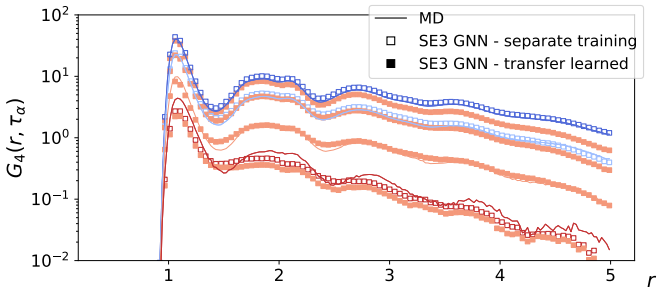


Figure 2.13: **Spatial dynamical correlations.** The function G_4 is computed on the true labels (MD, solid line) or on our predictions. Same color and marker coding as previous plot. Our models reproduce G_4 remarkably well ("separate training"), especially at low temperatures (blue), while the transfer-learned fields track the trends and orders of magnitude correctly as well.

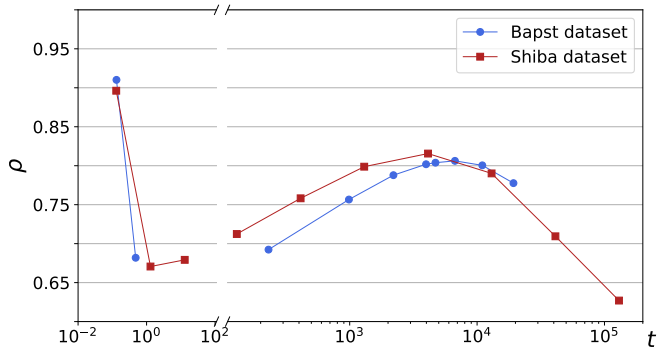


Figure 2.14: **Comparison of datasets.** The same models trained at temperature $T = 0.44$ on each dataset.

A complementary measure of the statistical quality of the predicted mobility field is given by the spatial correlation function of the mobility-related quantity $c_i(t)$:
$G_4(\mathbf{r}, t) = \frac{V}{N_A} \left\langle \sum_{i,j \in A} \tilde{c}_i(t)\tilde{c}_j(t)\delta(\mathbf{r} - \mathbf{r}_i(0) + \mathbf{r}_j(0)) \right\rangle$ where $\tilde{c}_i(t) = c_i(t) - \langle c(t) \rangle$. Our predictions reproduce it almost perfectly (see Figure 2.13).

Shiba vs Bapst In Bapst's dataset [BKGB+20], a timescale actually corresponds to a fixed value of the self-intermediate scattering function $F_k(t)$, so that different samples are measured at slightly different times. Equilibration is performed under an NPT thermostat, *i.e.* at constant pressure and temperature, *i.e.* the volume is varying (and thus the density as well).

In Shiba's dataset [SHSS23], equilibration is performed at constant volume and temperature (NVT) so that the density is exactly $\rho = 1.2$ in all samples and at all temperatures. Furthermore, sampling of the trajectories is performed at fixed times, not fixed $F_k(t)$.

In Figure 2.14 we report the performance of our main model (thermal positions + $E_{pot}(IS)$ inputs) for these two Kob-Anderson 3D datasets. Performances are shifted in time but otherwise rather comparable.

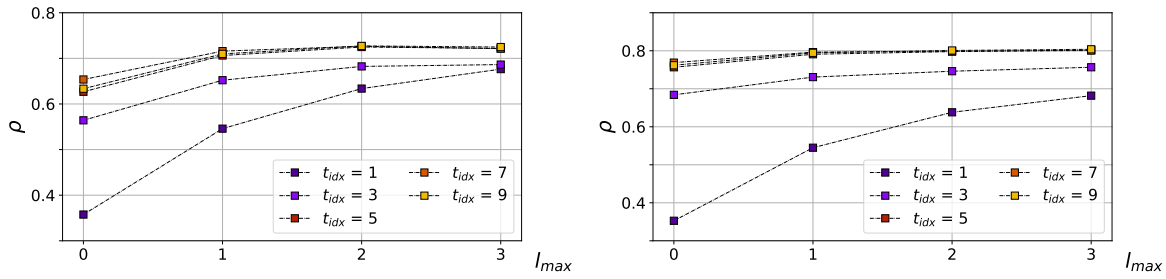


Figure 2.15: (*left*) $\boldsymbol{l_{max}}$ **ablation using thermal positions and no** $\boldsymbol{E_{pot}}$ **input.** A separate model was trained at $T = 0.44$, for each value of l_{max}. The colors correspond to the $n_{times} = 10$ timescales of mobility, with t_{idx} ranging from 0 to 9. For clarity, only some of these timescales are displayed here. The GNN of [BKGB+20] obtains $\rho \approx 0.65$ for the timescale 6, we are in this range when using only invariant features ($l_{max} = 0$). When using higher orders, we outperform it. (*right*) $\boldsymbol{l_{max}}$ **ablation using thermal positions combined with** $\boldsymbol{E_{pot}(IS)}$**.** Performance is overall higher, the relative gain from using $l > 0$ is less pronounced, probably because $E_{pot}(IS)$ already provides equivalent information.

Ablation Studies Here we display the ablation studies, that outline which are the key elements of our model. We also report the learning curve (ablation on training set size).

All our results rely on the embedding of the input data into the Spherical Harmonics basis and on the built-in equivariance of convolution layers. One may expect that a large cutoff rotational order l_{max} is needed. Here we show that actually, going from $l_{max} = 0$ to $l_{max} = 1$ is the most critical step. We build architectures that vary only by their l_{max} value and measure the performance ρ in each, as shown in Figure 2.15. The biggest gap in performance is indeed observed between purely isotropic, scalar features ($l_{max} = 0$) versus directional ones ($l_{max} = 1$). We notice as well that short timescales require higher rotational order and the performance indeed has not saturated for them. One possible interpretation is that the network has to learn inter-atomic forces to describe the dynamics at short times, and that directional information is more relevant in that case. Further increasing l_{max} provides a finer rotational order resolution, but we observe that the accuracy tends to saturate. We cannot go above $l_{max} = 3$ due to memory constraints: as the rotational order increases, the number of activations of neurons to keep track of grows exponentially with l_{max} (while it grows linearly with the number of edges, with the batch size, and with the size of the hidden layer in the radial MLP).

In Figure 2.16-left we present the performance of our multi-time model trained at temperature $T = 0.44$ for an increasing number L_{max} of equivariant convolution layers stacked in the architecture. While the short timescales seem to be saturated, the longer ones seem not: indeed, we'd expect increased accuracy if we increased L_{max} further. Note that there is a possibility of encountering over-smoothing effects with an increased number of layers.

In Figure 2.16-right, we present learning curves that illustrate the model's performance (multivariate setting) as a function of the number of samples in the training set for different timescales. The choice of the samples to include in the training set is performed in an incremental way: for each point of the curve new samples are added to the others already present, while the test set (80 samples) is kept constant. In this version we used early stopping with a validation set of 320 samples, however using the last epoch's model yields very similar result. In [JAB+23] we used the

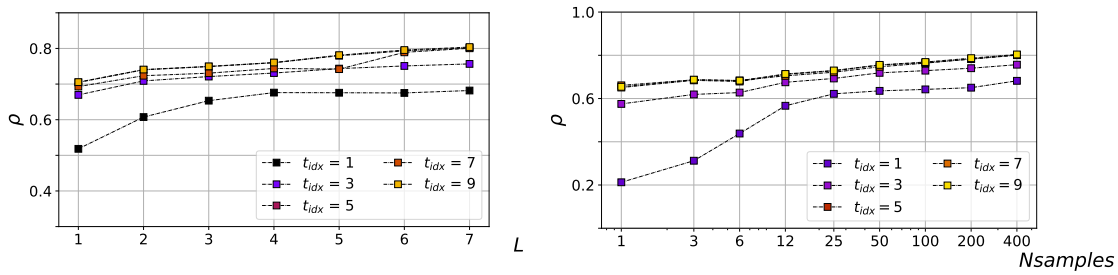


Figure 2.16: (*left*) $\boldsymbol{L_{max}}$ **ablation.** Color-code is the same as for the other ablation studies, but here we vary the number of convolution layers applied. Performance does not seem to saturate: one expects increased performance with more layers. (*right*) **Learning curve** Performance is already very high when training on a single sample: our network seems to resist well to overfitting. Here again, performance does not seem to saturate, more precisely it seems to increase logarithmically with train set size.

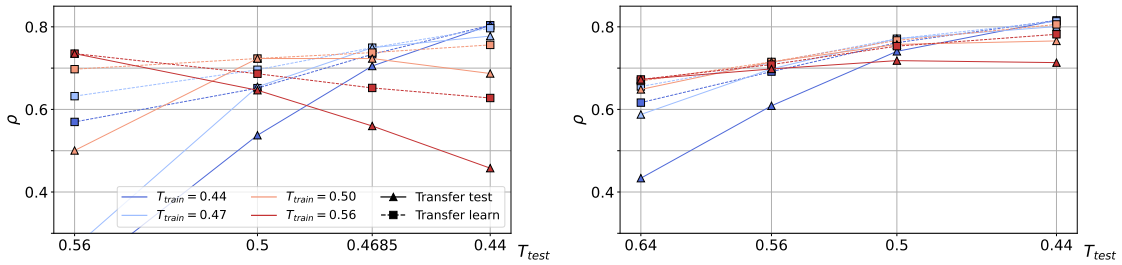


Figure 2.17: **Transfer-learning between different temperatures.** Each model is fully trained once at one state point (T) and tested ("transfer test") or fine-tuned on the remaining ones ("transfer learn"). The timescale of mobilities showed in the plot is $\tau = \tau_\alpha(T)$, but multi-times models were used. (*left*) Bapst' dataset, (*right*) Shiba's dataset. For each training temperature (color) two different experiments are performed: transfer test (square markers with dashed lines) and transfer learn (upper triangle with full line). This results in 8 curves per plot coming from all the combinations of colors and line-styles. The transfer learned-generalization on Shiba's dataset is almost indistinguishable from direct training, indicating excellent generalization power of our learnt representation.

last epoch's model and observe similar behavior. We emphasize that competitive performances are achieved already by using less than 1/4 of the available training set and meaningful prediction are obtained also when training the model on a single sample, contrary to what one would expect for a "deep" model like ours.

2.4.4 Temperature Generalization: Machine Learned Order Parameter

Here, we aim to advance the idea that deep learning can serve as a valuable tool for defining a structural order parameter in glassy systems. As introduced in Sec. 2.2, the primary goal in developing a microscopic theory for glasses is to find a structural order parameter that identifies amorphous order, which increases as the temperature decreases, and relates this growth to the dynamical length scale. Deep learning models provide a powerful framework to define such an order parameter. In practice, this involves designing a machine learning model, specifically a function $f_\theta(\{\mathbf{x}\})$, that takes the structure as input and is trained to predict local mobility (such as propensity). The result is a formally structure-based function, $f_\theta(\{\mathbf{x}\})$, capable of locating

"soft" and "hard" regions within the structure and displaying an increasing length scale corresponding to the dynamical heterogeneities. This function will demonstrate a sharp distinction between active and passive regions, with a noticeable change around T_g, resembling the behavior of a structural order parameter.

A counter-argument to this approach is that such an order parameter is not strictly structure-based, as it uses mobility as the training data. Critics may argue that, since "neural networks can overfit," the function $f_\theta(\{\mathbf{x}\})$ might simply track the mobility it was trained on, rather than capturing true structural features. Indeed, a complex network with millions of parameters, specialized to a particular temperature and timescale, could associate mobility variations with minor peculiarities specific to that temperature and timescale, potentially failing to generalize to others. Furthermore, if the function f_θ becomes too complex to interpret, it might not provide any deeper understanding of the underlying physics, rendering it unsatisfactory as an order parameter. In this view, even if a network achieves a perfect correlation ($\rho = 1$) in predicting mobility, it could be seen as merely a computational shortcut to predict iso-configurational displacement faster than Molecular Dynamics simulations, which could still be useful—for instance, in designing effective glass models, as shown in [ZXY+22].

However, we argue that a deep learning model, f_θ, should be viewed as a microscope that magnifies subtle structural variations within the material. Training the model to predict mobility is simply a means to extract relevant structural information, with the specifics of the training process being secondary. A true structural order parameter, $f(\{\mathbf{x}\})$, must be universally defined for a given system, irrespective of temperature. This translates into applying the same trained model f_θ across all temperatures for a given glass-former. A crucial test for determining whether the model captures relevant structural changes is to evaluate its ability to predict mobility and its spatial and temporal correlations, especially at temperatures other than the one used for training.

If the model successfully generalizes across different temperatures, this would directly challenge the criticism that "the neural network overfits." Such a transferability test was introduced several years ago [LBD+20, BKGB+20], and the idea of applying a trained model to different temperatures dates back to the original works on machine learning applied to glasses [SLRR14].

Transfer-testing Here we repeat this experiment and observe better temperature-generalization abilities of our network, as compared with the original work of Bapst [BKGB+20], as shown in Figure 2.17 (top part, label "Transfer test"). We also perform the same experiment using the more recent Shiba's dataset [SHSS23], showing even better temperature-generalization. This is a strong indication that our network learns the relevant subtle structural signatures rather than "overfitting" the dynamics separately at each temperature. We note that performance at a given temperature decreases as the training temperature goes further away from the test temperature (reading markers vertically). This can be attributed either to an increasing dissimilarity in the structures present, or also to a change in how these structures correlate with the dynamics at different temperatures. We also note an asymmetry in the performance drop between training at high temperature, testing at low (red line) or vice versa (blue line): the training at high temperature generalizes better, comparatively. In works based on SVM, the opposite was observed, and attributed to the noisy nature of high-temperature data. Here we do not seem to suffer from this noise, and attribute the increased generalizability to the larger diversity in input structures

observed and the broader range of propensities observed when training on high temperature data.

Transfer-learning Here we aim to push forward the idea of embracing the Deep Learning notion of learning *representations* and comment on the properties of our learned representation. Indeed, the convolution layers of our network effectively build an equivariant feature, $\mathbf{h}^{(L_{max})}$, that describes the local structure around each particle. The norm $|\mathbf{h}^{(L_{max})}|$ of these features is a list of 32 numbers (8 channels times 4 possible l values, $l = 0, 1, 2, 3$) that is decoded into mobility by 20 independent decoders. Thus, for any training temperature, the model must somehow pack the information about these 20 (non-independent) scalar values into the 32 components of $|\mathbf{h}^{(L_{max})}|$.

Here we further test the robustness of our model by evaluating the generalization ability of its underlying representation, $|\mathbf{h}^{(L_{max})}|$. If we consider $|\mathbf{h}^{(L_{max})}|$ as a general structural descriptor, it must be relevant at all temperatures. A simple way to test whether this structural measure captures the glass transition correctly is to verify if it tracks dynamics across different temperatures from the one used in training. Concretely, we train a representation $|\mathbf{h}^{(L_{max})}|$ by regressing labels at a given temperature, and then fine-tune only the decoders at other temperatures. The part of the network responsible for computing $|\mathbf{h}^{(L_{max})}|$ (most of the network) is frozen, so the fine-tuning reduces to learning the weights $\mathbf{w}$ of the decoders, as in Eq. 2.70, *i.e.*, only 32 values per timescale and per particle type. This transfer-learning strategy is central to Machine Learning and has been successful, *e.g.*, in computer vision. For example, a Convolutional Neural Network (CNN) is first trained on a dataset such as ImageNet with 1000 classes, and then the backbone of the network (all convolution layers) is frozen. The last layers that decode this representation into labels are re-trained on a different dataset, such as CIFAR10 or other natural images. This transfer-learning approach often yields better performance than training from scratch, particularly when fewer data are available for the final task [Ben12]. More importantly, the success of transfer learning indicates that the learned representation is more general than expected, suggesting that the backbone can be considered an advanced image pre-processing tool.

An application of transfer learning is few-shot learning [WYKN20], where a good representation is built from a large dataset (either labeled or through self-supervised learning), and a classifier is then trained using only a handful of examples (1 to 5 per class). In our case, $|\mathbf{h}^{(L_{max})}|$ effectively extracts structural features, or more precisely, detects patterns correlated with mobility.

We report the results of our transfer-learning experiment across temperatures in Figure 2.17. As expected, the performance (dashed lines) improves compared to transfer-testing (solid lines). Moving towards considering $|\mathbf{h}^{(L_{max})}|$ as a multidimensional equivalent of a structural order parameter, one could study how the coefficients $\mathbf{w}_{A,\tau_\alpha}$ depend on the target temperature, and, for example, attempt to fit them with interpretable functions of temperature. It is worth noting that among the components of $\mathbf{w}_{A,\tau_\alpha}$, most vary monotonously with temperature, particularly the $l = 0$ components (which dominate the total). We leave a deeper investigation of these coefficients for future work.

As an additional test of the robustness of our representation, we also report the transfer-learned estimates of χ_4^s in Figure 2.12, which show larger discrepancies compared to those trained at each

temperature, but still track the trends observed in the data (similarly in Figure 2.13). Note that this transfer-learned χ_4^s reflects structural heterogeneity since our input is purely structural, based on a single set of descriptors (the representation $|\mathbf{h}^{(L_{max})}|$). To our knowledge, this is the first time that a unique set of descriptors has been shown to display such large structural fluctuations across temperatures and timescales. For clarity, we do not show the transfer-test results (they typically perform slightly worse).

2.4.5 Future directions

Here we performed a non-exhaustive but thorough architecture search and found no performance gain when attempting several intuitive improvements for the network, such as increasing l_{max}, increasing the decoders' complexity, introducing bottleneck layers (by reducing l_{max} or the number of channels), using attention mechanisms as in [LS22, LWDS23, HLLZ+21], or assigning channels to specific bond types. This list of negative results does not preclude us from suggesting further improvements for enhanced performance, either for our model or others, which we leave for future work.

Potential improvements include:

- Fully leveraging the equivariant properties of the network to predict directly the displacement vector (comprising three components in 3D space), rather than just its scalar magnitude; this requires mobility to be computed from single-run positions rather than the iso-configurational average, which may yield non-physical directional displacements. Incorporating directional features from [LBVP22] could further enhance the extraction of relevant information.

- Decoding various timescales using a single timescale-aware decoder, similar to FiLM [PSDV+18] (conditioning the decoder with an embedding of the timescale, as proposed in [GB22], allowing for a single final decoder).

- Training the backbone on several temperatures simultaneously, with separate decoders for each temperature (possibly incorporating the previous idea to create a decoder that is both timescale-aware and temperature-aware).

- As in [JBB23], adding non-local quantities as additional target labels (i.e., adding terms in the loss function), such as global correlation functions evaluated at specific lengths (computed for the entire sample, resulting in a graph-wide target), or the local variance of the mobility (variance of the target label within a node's neighborhood). This could improve prediction quality, particularly in terms of spatio-temporal correlations, addressing over-smoothing—a known issue in GNNs.

- Using coarse-grained mobility measures as target labels. These have been shown to be more structure-dependent [BJ07] and display stronger correlations with simple structural descriptors [OKK23]. Reducing label noise may improve precision and potentially maximize performance.

Self-supervised learning offers a possible solution to the issue that our structural features are trained using dynamical data (as labels). Here, we propose a few self-supervised strategies:

- Denoising: Adding nonphysical noise to thermal (or quenched) positions and asking the network to denoise the input.
- Predicting only known quantities, such as E_{pot}, or quenched/thermal positions from thermal/quenched inputs.

Finally, it is important to emphasize that steerable convolutions are not the only available approach to achieve equivariance. Several alternative frameworks have emerged in recent years. These include leveraging Clifford Algebra for efficient handling of rotations and reflections [RBF23], employing Frame Averaging to adapt existing networks to be invariant or equivariant [PAB+21], and utilizing continuous Fourier analysis, which combines the benefits of group convolutions and steerable convolutions for greater flexibility [ZG24a, ZG24b]. These alternatives open up interesting possibilities for future research.

2.5 Conclusion

In this chapter, we introduced the concept of supercooled liquids and explored their phenomenology, briefly mentioning the theoretical approaches used to study them. We demonstrated that a comprehensive theoretical framework is still missing, one that can relate the growth of a structural order to the dramatic slowdown of relaxation times and the emergence of dynamic heterogeneities. We then explored ML approaches as useful tools for extracting structural features whose subtle variations underlie the changes in relaxation dynamics. This is the context in which our work was situated: utilizing advanced ML methods, particularly deep learning approaches, to design complex features that go beyond the capabilities of manually designed studies.

We also used this complex task as an opportunity to thoroughly investigate modern architectures, pushing them to their limits and working towards novel designs. In this chapter, we reviewed in detail the theoretical foundations of Equivariant Neural Networks and presented an adaptation of recent rotation-equivariant architectures, such as NequIP [BMS+22], for modeling glassy liquids. Specifically, **we had to adapt these architectures to handle large graphs** (4096 nodes) and **address our particular task of multi-variate node regression**. We also introduced some ideas specific to glasses: inspired by recent works on ML for glasses [JBB23, ASF23], **we combined information from thermal positions and their quenched counterparts, using the local potential energy of quenched positions as input, thereby boosting our network's performance**.

We compared our model with two families of architectures. On one hand, **compared to deep learning approaches**, particularly GNN models that are neither equivariant nor invariant [BKGB+20, SHSS23], **our SE(3)-equivariant architecture outperforms them with significantly fewer parameters, as soon as we use strictly equivariant features** ($l_{max} > 0$). In other words, we prove the usefulness of equivariance. On the other hand, **compared to shallow learning techniques** [BSF21, ASF23, JBB23] that use expert features as input (such as invariant features and local potential), **our deep network performs better, especially when enriched with combined information from thermal and quenched positions**. The deeper the network, the better the performance, particularly for longer timescales, suggesting that dynamics becomes increasingly non-local as time progresses.

Moreover, we emphasized the importance of building a robust representation: pure performance measured by the correlation of our predictions with the ground truth mobility is a means to an end, not the end goal itself. What truly matters is whether our representation of the local structure allows us to deduce physical insights. Our good correlation ρ, the very good fit of G_4, and the acceptable trends in predicted χ_4 all indicate that we have built a strong representation. **We are able to capture the local mobility field as well as its spatial and temporal correlations**.

Most importantly, the fact that a **representation learned at a given temperature generalizes well to other temperatures** suggests that this representation is more than just a learned structural descriptor: it approaches the concept of an acceptable structural order parameter. This generalization power largely stems from our use of an equivariant representation and is reinforced by our approach of regressing all particle types and timescales simultaneously, using a single backbone representation, with the predictions differing only in the final decoder.

Conclusion

This thesis aimed to underline the value of collaboration between Statistical Physics and Machine Learning, demonstrating how each field can enrich the other. The compelling nature of this intersection is highlighted not only by the increasing number of recent publications on the topic, but also by two recent Nobel Prizes in Physics. The 2024 Prize was awarded to John Hopfield for his pioneering work on Neural Networks rooted in Statistical Physics, and the 2021 Prize to Giorgio Parisi for his groundbreaking contribution to the understanding of complex physical systems, among which ML models.

In this work, I addressed two relevant challenges for these disciplines: the problem of class imbalance in supervised learning and the prediction of dynamical heterogeneities in supercooled liquids. While these topics may initially appear unrelated, they share a unifying theme: the exploration of complex, interacting systems using both theoretical and computational approaches.

The first part of this thesis explored the problem of class imbalance, studying an analytically tractable model, the teacher-student perceptron, through Statistical Physics tools. Building on this paradigmatic set-up, I modeled the input data to reproduce class imbalance of the Anomaly Detection type, where the imbalance is intrinsic to the problem rather than a result of the data collection process. This framework makes it possible to clarify the role of training imbalance (originating from data collection) relative to the intrinsic imbalance. Additionally, the framework provides a straightforward way to interpret the impact of imbalance, distinguishing "good" models from "bad" ones and identifying the key factors thanks to an energy-entropy interplay argument. At the same time, this framework validates the reliability of various performance metrics commonly used in empirical settings, ultimately identifying balanced accuracy as the most effective metric. Interestingly, this simple model reveals highly non-trivial behavior: contrary to commonly accepted wisdom in the field, a balanced dataset often leads to suboptimal performance. This is just an initial step towards fully understanding the mechanisms at play in supervised learning under class imbalance. There are many promising directions for future research, including modeling more complex datasets with inherent structure (correlations) and using more sophisticated models, such as kernel machines or deep architectures, to better simulate practical scenarios.

The second part of this thesis focused on predicting dynamical heterogeneities in supercooled

liquids. By utilizing advanced Machine Learning models, specifically Roto-Translational Equivariant Neural Networks, I was able to predict the mobility field with high accuracy based on the static structure of particles in numerical simulations of glass formers. In these materials, the mobility field exhibits spatial heterogeneity, characterized by distinct 'slow' and 'fast' regions. My model successfully captured the spatial correlations of such field and demonstrated temperature transferability, indicating that it had effectively learned a robust representation of the static structure. This representation suggests the existence of an amorphous order that grows as the temperature decreases and matches the growing length scale of dynamical observables, such as the typical size of 'fast' and 'slow' domains. While the model itself may not be fully interpretable, its emphasis on directional information implies the importance of incorporating and combining vectorial features of atomic packings, beyond simply rotationally invariant features, in identifying amorphous order. Future research in this area could aim to improve the model's accuracy and transferability to reliably detect structural features. Additionally, distilling information from the model could help to pinpoint the key physical observables that underlie the mechanism of dynamical slowdown in structural glasses.

The challenges tackled in this thesis illustrate the potential benefits of combining these domains: in one case theoretical methods from Statistical Physics provided a precise framework to characterize a Machine Learning problem, while in the other, ML tools were employed to effectively detect structural features in a physical model that elude traditional handcrafted descriptors. These are just two of the many examples emerging in recent years which hold promise for advancing our understanding of complex systems.

References

[AAA+19] Kim Albertsson, Piero Altoe, Dustin Anderson, John Anderson, Michael Andrews, Juan Pedro Araque Espinosa, Adam Aurisano, Laurent Basara, Adrian Bevan, Wahid Bhimji, Daniele Bonacorsi, Bjorn Burkle, Paolo Calafiura, Mario Campanelli, Louis Capps, Federico Carminati, Stefano Carrazza, Yi fan Chen, Taylor Childers, Yann Coadou, Elias Coniavitis, Kyle Cranmer, Claire David, Douglas Davis, Andrea De Simone, Javier Duarte, Martin Erdmann, Jonas Eschle, Amir Farbin, Matthew Feickert, Nuno Filipe Castro, Conor Fitzpatrick, Michele Floris, Alessandra Forti, Jordi Garra-Tico, Jochen Gemmler, Maria Girone, Paul Glaysher, Sergei Gleyzer, Vladimir Gligorov, Tobias Golling, Jonas Graw, Lindsey Gray, Dick Greenwood, Thomas Hacker, John Harvey, Benedikt Hegner, Lukas Heinrich, Ulrich Heintz, Ben Hooberman, Johannes Junggeburth, Michael Kagan, Meghan Kane, Konstantin Kanishchev, Przemysław Karpiński, Zahari Kassabov, Gaurav Kaul, Dorian Kcira, Thomas Keck, Alexei Klimentov, Jim Kowalkowski, Luke Kreczko, Alexander Kurepin, Rob Kutschke, Valentin Kuznetsov, Nicolas Köhler, Igor Lakomov, Kevin Lannon, Mario Lassnig, Antonio Limosani, Gilles Louppe, Aashrita Mangu, Pere Mato, Narain Meenakshi, Helge Meinhard, Dario Menasce, Lorenzo Moneta, Seth Moortgat, Mark Neubauer, Harvey Newman, Sydney Otten, Hans Pabst, Michela Paganini, Manfred Paulini, Gabriel Perdue, Uzziel Perez, Attilio Picazio, Jim Pivarski, Harrison Prosper, Fernanda Psihas, Alexander Radovic, Ryan Reece, Aurelius Rinkevicius, Eduardo Rodrigues, Jamal Rorie, David Rousseau, Aaron Sauers, Steven Schramm, Ariel Schwartzman, Horst Severini, Paul Seyfert, Filip Siroky, Konstantin Skazytkin, Mike Sokoloff, Graeme Stewart, Bob Stienen, Ian Stockdale, Giles Strong, Wei Sun, Savannah Thais, Karen Tomko, Eli Upfal, Emanuele Usai, Andrey Ustyuzhanin, Martin Vala, Justin Vasel, Sofia Vallecorsa, Mauro Verzetti, Xavier Vilasís-Cardona, Jean-Roch Vlimant, Ilija Vukotic, Sean-Jiun Wang, Gordon Watts, Michael Williams, Wenjing Wu, Stefan Wunsch, Kun Yang, and Omar Zapata. Machine learning in high energy physics community white paper, 2019.

[ABJ24] A. Altieri and M. Baity-Jesi. An introduction to the theory of spin glasses. In Chakraborty and Tapash, editors, *Encyclopedia of Condensed Matter Physics, 2e*, volume 2, pages 361–370. Elsevier, Oxford, 2024.

[AH17] Shin Ando and Chun-Yuan Huang. Deep Over-sampling Framework for Classifying Imbalanced Data, July 2017. arXiv:1704.07515 [cs, stat].

[AHY11] Tiago A Almeida, José María G Hidalgo, and Akebo Yamakami. Contributions to the study of sms spam filtering: new collection and results. In *Proceedings of the 11th ACM symposium on Document engineering*, pages 259–262, 2011.

[ALBB20] Francesco Arceri, François P. Landes, Ludovic Berthier, and Giulio Biroli. *A Statistical Mechanics Perspective on Glasses and Aging*, pages 1–68. Springer Berlin Heidelberg, Berlin, Heidelberg, 2020.

[Ang97] C. A. Angell. Formation of glasses from liquids and biopolymers. *Journal of Research of the National Institute of Standards and Technology*, 102:171–185, 1997.

[ASF23] Rinske M. Alkemade, Frank Smallenburg, and Laura Filion. Improving the prediction of glassy dynamics by pinpointing the local cage. January 2023. arXiv:2301.13106 [cond-mat].

[BB04] J.-P. Bouchaud and G. Biroli. On the adam-gibbs-kirkpatrick-thirumalai-wolynes scenario for the viscosity increase in glasses. *The Journal of Chemical Physics*, 121:7347–7354, 2004.

[BBB+07] L. Berthier, G. Biroli, J.-P. Bouchaud, W. Kob, K. Miyazaki, and D. R. Reichman. Spontaneous and induced dynamic fluctuations in glass formers. I. General results and dependence on ensemble and dynamics. *The Journal of Chemical Physics*, 126(18):184503, May 2007.

[BBB+11] Ludovic Berthier, Giulio Biroli, Jean-Philippe Bouchaud, Luca Cipelletti, and Wim van Saarloos. *Dynamical heterogeneities in glasses, colloids, and granular media*, volume 150. OUP Oxford, 2011.

[BCGV08] G.and Bouchaud J. P. Biroli, A Cavagna, T. S. Grigera, and P. Verrocchio. Thermodynamic signature of growing amorphous order in glass-forming liquids. *Nature Physics*, 4:771–775, 2008.

[BCNO16] Ludovic Berthier, Daniele Coslovich, Andrea Ninarello, and Misaki Ozawa. Equilibrium sampling of hard spheres up to the jamming density and beyond. *Physical review letters*, 116(23):238002, 2016.

[BCP20] Blake Bordelon, Abdulkadir Canatar, and Cengiz Pehlevan. Spectrum dependent learning curves in kernel regression and wide neural networks. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1024–1034. PMLR, 13–18 Jul 2020.

[Bek21] Erik J. Bekkers. An introduction to equivariant convolutional neural networks for continuous groups, 2021. Available at `https://uvagedl.github.io/GroupConvLectureNotes.pdf`.

[Ben12] Yoshua Bengio. Deep learning of representations for unsupervised and transfer learning. In *Proceedings of ICML workshop on unsupervised and transfer learning*, pages 17–36. JMLR Workshop and Conference Proceedings, 2012.

[BHvdP+21] Johannes Brandstetter, Rob Hesselink, Elise van der Pol, Erik J Bekkers, and Max Welling. Geometric and physical quantities improve e (3) equivariant message passing. In *International Conference on Learning Representations*, 2021.

[BJ07] Ludovic Berthier and Robert L. Jack. Structure and dynamics in glass-formers: predictability at large length scales. *Physical Review E*, 76(4):041509, October 2007.

[BJ12] Ludovic Berthier and Robert L. Jack. Static point-to-set correlations in glass-forming liquids. *Physical Review E*, 85(1):011102, 2012.

[BKG+06] Erik Bitzek, Pekka Koskinen, Franz Gähler, Michael Moseler, and Peter Gumbsch. Structural relaxation made simple. *Phys. Rev. Lett.*, 97:170201, Oct 2006.

[BKGB+20] V. Bapst, T. Keck, A. Grabska-Barwińska, C. Donner, E. D. Cubuk, S. S. Schoenholz, A. Obika, A. W. R. Nelson, T. Back, D. Hassabis, and P. Kohli. Unveiling the predictive power of static structure in glassy systems. *Nature Physics*, 16(4):448–454, April 2020.

[BKS+22] Ilyes Batatia, David P Kovacs, Gregor Simm, Christoph Ortner, and Gábor Csányi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. *Advances in Neural Information Processing Systems*, 35:11423–11436, 2022.

[BKVT23] Tina Behnia, Ganesh Ramachandra Kini, Vala Vakilian, and Christos Thrampoulidis. On the Implicit Geometry of Cross-Entropy Parameterizations for Label-Imbalanced Data. 2023.

[BM01] Giulio Biroli and Marc Mézard. Lattice glass models. *Physical Review Letters 2001-dec 26 vol. 88 iss. 2*, 88, dec 2001.

[BM19] J. Barbier and N. Macris. The adaptive interpolation method: a simple scheme to prove replica formulas in bayesian inference. *Probability Theory and Related Fields*, 2019.

[BMAM+20] Emanuele Boattini, Susana Marín-Aguilar, Saheli Mitra, Giuseppe Foffi, Frank Smallenburg, and Laura Filion. Autonomously revealing hidden local structures in supercooled liquids. *Nature Communications*, 11(1):5479, December 2020.

[BMS+22] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1):2453, 2022.

[Bot99] Léon Bottou. On-line Learning and Stochastic Approximations. In David Saad, editor, *On-Line Learning in Neural Networks*, pages 9–42. Cambridge University Press, 1 edition, January 1999.

[BR22] Ludovic Berthier and David R. Reichman. Modern computational studies of the glass transition, August 2022. arXiv:2208.02206 [cond-mat].

[BSF21] Emanuele Boattini, Frank Smallenburg, and Laura Filion. Averaging local structure to predict the dynamic propensity in supercooled liquids. *Physical Review Letters*, 127(8):088007, August 2021. arXiv: 2105.05921.

[CABJ20] Matthew R. Carbone, Valerio Astuti, and Marco Baity-Jesi. Effective traplike activated dynamics in a continuous landscape. *Phys. Rev. E*, 101:052304, May 2020.

[Cav09] Andrea Cavagna. Supercooled Liquids for Pedestrians. *Physics Reports*, 476(4-6):51–124, June 2009. arXiv: 0903.4264.

[CBHK02] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16:321–357, June 2002.

[CC05] Tommaso Castellani and Andrea Cavagna. Spin-Glass Theory for Pedestrians. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(05):P05012, May 2005. arXiv: cond-mat/0505032.

[CCC+19] Giuseppe Carleo, Ignacio Cirac, Kyle Cranmer, Laurent Daudet, Maria Schuld, Naftali Tishby, Leslie Vogt-Maranto, and Lenka Zdeborová. Machine learning and the physical sciences. *Rev. Mod. Phys.*, 91:045002, Dec 2019.

[CDS23] Giovanni Catania, Aurélien Decelle, and Beatriz Seoane. The Copycat Perceptron: Smashing Barriers Through Collective Learning, August 2023. arXiv:2308.03743 [cond-mat].

[CG10] David Chandler and Juan P Garrahan. Dynamics on the way to forming glass: Bubbles in space-time. *Annual review of physical chemistry*, 61:191–217, 2010.

[CGV07] A Cavagna, T. S. Grigera, and P. Verrocchio. Mosaic multi-state scenario vs. one-state description of supercooled liquids. *Phy. Rev. Lett.*, 98:187801, 2007.

[CHS93] Andrea Crisanti, Heinz Horner, and H J Sommers. The spherical p-spin interaction spin-glass model: the dynamics. *Zeitschrift für Physik B Condensed Matter*, 92:257–271, 1993.

[CIS+17] E. D. Cubuk, R. J. S. Ivancic, S. S. Schoenholz, D. J. Strickland, A. Basu, Z. S. Davidson, J. Fontaine, J. L. Hor, Y.-R. Huang, Y. Jiang, N. C. Keim, K. D. Koshigan, J. A. Lefever, T. Liu, X.-G. Ma, D. J. Magagnosc, E. Morrow, C. P. Ortiz, J. M. Rieser, A. Shavit, T. Still, Y. Xu, Y. Zhang, K. N. Nordstrom, P. E. Arratia, R. W. Carpick, D. J. Durian, Z. Fakhraai, D. J. Jerolmack, Daeyeon Lee, Ju Li, R. Riggleman, K. T. Turner, A. G. Yodh, D. S. Gianola, and Andrea J. Liu. Structure-property relationships from universal signatures of plasticity in disordered solids. *Science*, 358(6366):1033–1037, November 2017. ISBN: 3487716170192.

[CJP22] Daniele Coslovich, Robert L Jack, and Joris Paret. Dimensionality reduction of local structure in glassy binary mixtures. *The Journal of Chemical Physics*, 157(20), 2022.

[CKZ23] Hugo Cui, Florent Krzakala, and Lenka Zdeborova. Bayes-optimal learning of deep random networks of extensive-width. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 6468–6521. PMLR, 23–29 Jul 2023.

[CLB+21] Rahul N Chacko, François P Landes, Giulio Biroli, Olivier Dauchot, Andrea J Liu, and David R Reichman. Elastoplasticity mediates dynamical heterogeneity below the mode coupling temperature. *Physical Review Letters*, 127(4):048002, 2021.

[CM17] Juan Carrasquilla and Roger G. Melko. Machine learning phases of matter. *Nature Physics*, 13(5):431–434, May 2017.

[CMV+23] Elisabetta Cornacchia, Francesca Mignacco, Rodrigo Veiga, Cédric Gerbelot, Bruno Loureiro, and Lenka Zdeborová. Learning curves for the multi-class teacher-student perceptron. *Machine Learning: Science and Technology*, 4(1):015019, March 2023. arXiv:2203.12094 [cond-mat, stat].

[Cos11] Daniele Coslovich. Locally preferred structures and many-body static correlations in viscous liquids. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 83(5):1–8, 2011. arXiv: 1102.5663.

[CSMT18] Stefan Chmiela, Huziel E. Sauceda, Klaus-Robert Müller, and Alexandre Tkatchenko. Towards exact molecular dynamics simulations with machine-learned force fields. *Nature Communications*, 9(1):3887, September 2018.

[CSR+15] E. D. Cubuk, S. S. Schoenholz, J. M. Rieser, B. D. Malone, J. Rottler, D. J. Durian, E. Kaxiras, and A. J. Liu. Identifying structural flow defects in disordered solids using machine-learning methods. *Physical Review Letters*, 114(10):1–5, 2015. arXiv: 1409.6820.

[CW16a] Taco Cohen and Max Welling. Group equivariant convolutional networks. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 2990–2999, New York, New York, USA, 20–22 Jun 2016. PMLR.

[CW16b] Taco S. Cohen and Max Welling. Steerable CNNs, December 2016. arXiv:1612.08498 [cs, stat].

[CWCK+10] R. Candelier, A. Widmer-Cooper, J. K. Kummerfeld, O. Dauchot, G. Biroli, P. Harrowell, and D. R. Reichman. Spatiotemporal hierarchy of relaxation events, dynamical heterogeneities, and structural reorganization in a supercooled liquid. *Physical Review Letters*, 105(13):135702, September 2010. arXiv: 0912.0193 ISBN: 0031-9007\n1079-7114.

[CYZ+19] Chi Chen, Weike Ye, Yunxing Zuo, Chen Zheng, and Shyue Ping Ong. Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals. *Chemistry of Materials*, 31(9):3564–3572, May 2019.

[dAT78] JRL de Almeida and DJ Thouless. Stability of the sherrington-kirkpatrick solution of a spin glass model. *Journal of Physics A: Mathematical and General*, 11(5):983–990, 1978.

[Daw02] Kenneth A. Dawson. The glass paradigm for colloidal glasses, gels, and other arrested states driven by attractive interactions. *Current Opinion in Colloid and Interface Science 2002-aug vol. 7 iss. 3-4*, 7, aug 2002.

[DFGP02] Claudio Donati, Silvio Franz, Sharon C. Glotzer, and Giorgio Parisi. Theory of non-linear susceptibility and correlation length in glasses and liquids. *Journal of Non-Crystalline Solids 2002-sep vol. 307-310 iss. none*, 307-310, sep 2002.

[DGP99] Claudio Donati, Sharon C. Glotzer, and Peter H. Poole. Growing spatial correlations of particle displacements in a simulated liquid on cooling toward the glass transition. *Phys. Rev. Lett.*, 82:5064–5067, Jun 1999.

[DKL+23] Yatin Dandi, Florent Krzakala, Bruno Loureiro, Luca Pesce, and Ludovic Stephan. How two-layer neural networks learn, one (giant) step at a time, 2023.

[dRBK20] Stéphane d'Ascoli, Maria Refinetti, Giulio Biroli, and Florent Krzakala. Double trouble in double descent : Bias and variance(s) in the lazy regime, 2020.

[DS01] P. G. Debenedetti and F. H. Stillinger. Supercooled liquids and the glass transition. *Nature*, 410:259–267, 2001.

[dSB21] Stéphane d'Ascoli, Levent Sagun, and Giulio Biroli. Triple descent and the two kinds of overfitting: where and why do they appear?*. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124002, December 2021.

[DV01] D Dalmazi and JJM Verbaarschot. The replica limit of unitary matrix integrals. *Nuclear Physics B*, 592(3):419–433, 2001.

[EA75] S. F. Edwards and P. W. Anderson. Theory of spin glasses. *Journal of Physics F: Metal Physics*, 5:965, 1975.

[Eng12] A Engel. *Statistical Mechanics of Learning*. Cambridge University Press, 2012.

[FBJL23] Emanuele Francazi, Marco Baity-Jesi, and Aurelien Lucchi. Characterizing the effect of class imbalance on the learning dynamics. *International Conference of Machine Learning (ICML)*, 2023.

[FL19] Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[FM93] J F Fontanari and R Meir. The statistical mechanics of the Ising perceptron. *Journal of Physics A: Mathematical and General*, 26(5):1077–1089, March 1993.

[FML+02] Giuseppe Foffi, Gavin D. McCullagh, Aonghus Lawlor, Emanuela Zaccarelli, Kenneth A. Dawson, Francesco Sciortino, Piero Tartaglia, Davide Pini, and George Stell. Phase equilibria and glass transition in colloidal systems with short-ranged

attractive interactions: Application to protein crystallization. *Physical Review E 2002-mar 01 vol. 65 iss. 3*, 65, mar 2002.

[FWFW20] Fabian B Fuchs, Daniel E Worrall, Volker Fischer, and Max Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, pages 1970–1981, 2020.

[Gar87] E Gardner. Maximum Storage Capacity in Neural Networks. *Europhysics Letters (EPL)*, 4(4):481–485, August 1987.

[Gar09] Crispin Gardiner. *Stochastic Methods: A Handbook for the Natural and Social Sciences*. Number 0172-7389 in Springer Series in Synergetics. Springer, Berlin, Heidelberg, 2009.

[GAS+19] Sebastian Goldt, Madhu S. Advani, Andrew M. Saxe, Florent Krzakala, and Lenka Zdeborová. Generalisation dynamics of online learning in over-parameterised neural networks, 2019.

[GB22] Jayesh K Gupta and Johannes Brandstetter. Towards multi-spatiotemporal-scale generalized pde modeling. *arXiv preprint arXiv:2209.15616*, 2022.

[GD89] E Gardner and B Derrida. Three unfinished works on the optimal storage capacity of networks. *Journal of Physics A: Mathematical and General*, 22(12):1983–1994, June 1989.

[GJL+07] J. P. Garrahan, R. L. Jack, V. Lecomte, E. Pitard, K. van Duijvendijk, and F. van Wijland. Dynamical First-Order Phase Transition in Kinetically Constrained Models of Glasses. *Physical Review Letters*, 98(19):195702, May 2007.

[GLZ+23] X. Gong, H. Li, N. Zou, R. Xu, W. Duan, and Y. Xu. General framework for e (3)-equivariant neural network representation of density functional theory hamiltonian. *Nature Communications*, 14:2848, 2023.

[GP01] Tomás S Grigera and Giorgio Parisi. Fast monte carlo algorithm for supercooled soft spheres. *Physical Review E*, 63(4):045102, 2001.

[GPAM+14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[GSd+19] Mario Geiger, Stefano Spigler, Stéphane d'Ascoli, Levent Sagun, Marco Baity-Jesi, Giulio Biroli, and Matthieu Wyart. Jamming transition as a paradigm to understand the loss landscape of deep neural networks. *Physical Review E*, 100(1), July 2019.

[GSM+22] Mario Geiger, Tess Smidt, Alby M., Benjamin Kurt Miller, Wouter Boomsma, Bradley Dice, Kostiantyn Lapchevskyi, Maurice Weiler, Michał Tyszkiewicz, Simon Batzner, Dylan Madisetti, Martin Uhrin, Jes Frellsen, Nuri Jung, Sophia Sanborn, Mingjian Wen, Josh Rackers, Marcel Rød, and Michael Bailey. Euclidean neural networks: e3nn. April 2022.

[GSTC96] P. Gallo, F. Sciortino, P. Tartaglia, and S.-H. Chen. Slow dynamics of water molecules in supercooled states. *Physical Review Letters 1996-apr 08 vol. 76 iss. 15*, 76, apr 1996.

[Gyo90] Géza Gyorgyi. First-order transition to perfect generalization in a neural network with binary synapses. *Physical Review A*, 41(12):7097–7100, June 1990.

[HLLZ+21] Michael J Hutchinson, Charline Le Lan, Sheheryar Zaidi, Emilien Dupont, Yee Whye Teh, and Hyunjik Kim. Lietransformer: Equivariant self-attention for lie groups. In *International Conference on Machine Learning*, pages 4533–4543. PMLR, 2021.

[Hop82] J J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982.

[HSD+21] Weihua Hu, Muhammed Shuaibi, Abhishek Das, Siddharth Goyal, Anuroop Sriram, Jure Leskovec, Devi Parikh, and C. Lawrence Zitnick. ForceNet: A Graph Neural Network for Large-Scale Quantum Calculations. *arXiv:2103.01436 [cs]*, March 2021. arXiv: 2103.01436.

[HZRS15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[IAS23] Clemens Isert, Kenneth Atz, and Gisbert Schneider. Structure-based drug design with geometric deep learning. *Current Opinion in Structural Biology*, 79:102548, 2023.

[IS15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.

[JAB+23] Gerhard Jung, Rinske M Alkemade, Victor Bapst, Daniele Coslovich, Laura Filion, François P Landes, Andrea Liu, Francesco Saverio Pezzicoli, Hayato Shiba, Giovanni Volpe, et al. Roadmap on machine learning glassy liquids. *arXiv preprint arXiv:2311.14752*, 2023.

[JBB23] Gerhard Jung, Giulio Biroli, and Ludovic Berthier. Predicting dynamic heterogeneity in glass-forming liquids by physics-inspired machine learning. *Physical Review Letters*, 130(23):238202, 2023.

[JEP+21] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596:583–589, 2021.

[JKA+17] Stanislaw Jastrzebski, Zachary Kenton, Devansh Arpit, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos Storkey. Three factors influencing minima in sgd. *arXiv:1711.04623*, 2017.

[JMG+98] Gregory Johnson, Andrew I. Mel'cuk, Harvey Gould, W. Klein, and Raymond D. Mountain. Molecular-dynamics study of long-lived structures in a fragile glass-forming liquid. *Phys. Rev. E*, 57:5707–5718, May 1998.

[Joh02] William Johnson. Bulk amorphous metal—an emerging engineering material. *JOM*, 54:40–43, 03 2002.

[JS02] Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6(5):429–449, November 2002.

[JTL22] Xiao Jiang, Zean Tian, and Kenli Li. Geometry-enhanced graph neural network for glassy dynamics prediction. November 2022. arXiv:2211.12832 [cond-mat].

[KA95a] W. Kob and H. C. Andersen. Testing mode-coupling theory for a supercooled binary lennard-jones mixture. i. the van hove correlation function. *Physical Review E*, 51:4626–4641, 1995.

[KA95b] W. Kob and H. C. Andersen. Testing mode-coupling theory for a supercooled binary lennard-jones mixture. ii. intermediate scattering function and dynamic susceptibility. *Physical Review E*, 52:4134–4153, 1995.

[KAE22] Firuz Kamalov, Amir F. Atiya, and Dina Elreedy. Partial Resampling of Imbalanced Data, July 2022. arXiv:2207.04631 [cs].

[KHM+21] Sreenath P Kyathanahally, Thomas Hardeman, Ewa Merz, Thea Bulas, Marta Reyes, Peter Isles, Francesco Pomati, and Marco Baity-Jesi. Deep learning classification of lake zooplankton. *Frontiers in microbiology*, page 3226, 2021.

[KIG11] Aaron S. Keys, Christopher R. Iacovella, and Sharon C. Glotzer. Characterizing Structure Through Shape Matching and Applications to Self-Assembly. *Annual Review of Condensed Matter Physics*, 2(1):263–285, March 2011.

[KMB+16] Steven Kearnes, Kevin McCloskey, Marc Berndl, Vijay Pande, and Patrick Riley. Molecular graph convolutions: moving beyond fingerprints. *Journal of Computer-Aided Molecular Design*, 30(8):595–608, August 2016.

[KMN+17] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima, February 2017. arXiv:1609.04836 [cs].

[Kob03] W. Kob. Supercooled liquids, the glass transition and computer simulations. In J.-L. Barrat, M. Feigelman, J. Kurchan, and J. Dalibard, editors, *Les Houches. Session LXXVII*, page 199. EDP Sciences, Les Ulis; Springer-Verlag, Berlin, Les Ulis, Berlin, 2003.

[KPOT21] Ganesh Ramachandra Kini, Orestis Paraskevas, Samet Oymak, and Christos Thrampoulidis. Label-Imbalanced and Group-Sensitive Classification under Overparameterization. 2021.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[LBD+20] François P. Landes, Giulio Biroli, Olivier Dauchot, Andrea J. Liu, and David R. Reichman. Attractive versus truncated repulsive supercooled liquids: The dynam-

ics is encoded in the pair correlation function. *Physical Review E*, 101(1):010602, January 2020. arXiv: 1906.01103.

[LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521:436–444, 2015.

[LBP+20] Maxime W. Lafarge, Erik J. Bekkers, Josien P. W. Pluim, Remco Duits, and Mitko Veta. Roto-translation equivariant convolutional networks: Application to histopathology image analysis. *CoRR*, abs/2002.08725, 2020.

[LBVP22] Matthias Lerbinger, Armand Barbot, Damien Vandembroucq, and Sylvain Patinet. Relevance of shear transformations in the relaxation of supercooled liquids. *Physical Review Letters*, 129(19):195501, 2022.

[LLC+24] Mingquan Liu, Chunyan Li, Ruizhe Chen, Dongsheng Cao, and Xiangxiang Zeng. Geometric deep learning for drug discovery. *Expert Systems with Applications*, 240:122498, 2024.

[LNC22] Tuan Le, Frank Noé, and Djork-Arné Clevert. Equivariant graph attention networks for molecular property prediction, 2022.

[LPCM24] Emanuele Loffredo, Mauro Pastore, Simona Cocco, and Rémi Monasson. Restoring balance: principled under/oversampling of data for optimal classification, May 2024. arXiv:2405.09535 [cond-mat].

[LS22] Yi-Lun Liao and Tess Smidt. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. In *The Eleventh International Conference on Learning Representations*, 2022.

[LWDS23] Yi-Lun Liao, Brandon Wood, Abhishek Das, and Tess Smidt. Equiformerv2: Improved equivariant transformer for scaling to higher-degree representations. *arXiv preprint arXiv:2306.12059*, 2023.

[MBJ+23] A. Musaelian, S. Batzner, A. Johansson, et al. Learning local equivariant representations for large-scale atomistic dynamics. *Nature Communications*, 14:579, 2023.

[MGRS23] Stefano Sarao Mannelli, Federica Gerace, Negar Rostamzadeh, and Luca Saglietti. Unfair geometries: exactly solvable data model with fairness implications, February 2023. arXiv:2205.15935 [cond-mat, stat].

[MJR+21] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment, July 2021. arXiv:2007.07314 [cs, stat].

[MKL+20] Francesca Mignacco, Florent Krzakala, Yue M Lu, Pierfrancesco Urbani, and Lenka Zdeborová. The Role of Regularization in Classification of High-dimensional Noisy Gaussian Mixture. 2020.

[MMN18] Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33), July 2018.

[MPV87] M. Mezard, G. Parisi, and M.A. Virasoro. *Spin Glass Theory And Beyond: An Introduction To The Replica Method And Its Applications*. World Scientific Lecture Notes In Physics. World Scientific Publishing Company, 1987.

[MS06] Andrea Montanari and Guilhem Semerjian. On the dynamics of the glass transition on bethe lattices. *Journal of statistical physics*, 124(1):103–189, 2006.

[Myd93] J A Mydosh. *Spin Glasses: An Experimental Introduction*. CRC Press, Taylor and Francis, 0 edition, 1993.

[NBC17] Andrea Ninarello, Ludovic Berthier, and Daniele Coslovich. Models and algorithms for the next generation of glass transition studies. *Physical Review X*, 7(2):021039, 2017.

[Nis01] Hidetoshi Nishimori. *Statistical physics of spin glasses and information processing: an introduction*. Number 111. Clarendon Press, 2001.

[Nob24] Nobel Prize Outreach AB. Computational protein design and protein structure prediction: Scientific background to the nobel prize in chemistry 2024, 2024. Available at `https://www.nobelprize.org/prizes/chemistry/2024/advanced-information/`.

[OKK23] Norihiro Oyama, Shihori Koyama, and Takeshi Kawasaki. What do deep neural networks find in disordered structures of glasses? *Frontiers in Physics*, 10:1320, 2023.

[PAB+21] Omri Puny, Matan Atzmon, Heli Ben-Hamu, Edward J. Smith, Ishan Misra, Aditya Grover, and Yaron Lipman. Frame averaging for invariant and equivariant network design. *CoRR*, abs/2110.03336, 2021.

[Pas22] M Pastore. Replicas in complex systems: applications to large deviations and neural networks. *pcteserver.mi.infn.it*, 2022.

[PCL24] Francesco Saverio Pezzicoli, Guillaume Charpiat, and François Pascal Landes. Rotation-equivariant graph neural networks for learning glassy liquids representations. *SciPost Physics*, 16(5):136, May 2024.

[PdON18] Michela Paganini, Luke de Oliveira, and Benjamin Nachman. Calogan: Simulating 3d high energy particle showers in multilayer electromagnetic calorimeters with generative adversarial networks. *Phys. Rev. D*, 97:014021, Jan 2018.

[PJC20] Joris Paret, Robert L. Jack, and Daniele Coslovich. Assessing the structural heterogeneity of supercooled liquids through community inference. *The Journal of Chemical Physics*, 152(14):144502, April 2020.

[PSDV+18] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[PVF16] Sylvain Patinet, Damien Vandembroucq, and Michael L. Falk. Connecting local yield stresses with plastic activity in amorphous solids. *Physical Review Letters*, 117(4):045501, 2016.

[RA98] R. Richert and C. A. Angell. Dynamics of glass-forming liquids. v. on the link between molecular dynamics and configurational entropy. *The Journal of Chemical Physics*, 108:9016–9026, 1998.

[RBF23] David Ruhe, Johannes Brandstetter, and Patrick Forré. Clifford group equivariant neural networks. *arXiv preprint arXiv:2305.11141*, 2023.

[RCBH20] Tal Reiss, Niv Cohen, Liron Bergman, and Yedid Hoshen. PANDA - adapting pretrained features for anomaly detection. *CoRR*, abs/2010.05903, 2020.

[RCH+11] P Ronhovde, S Chakrabarty, D Hu, M Sahu, KK Sahu, KF Kelton, NA Mauro, and Z Nussinov. Detecting hidden spatial and spatio-temporal structures in glasses and complex physical systems by multiresolution network clustering. *The European Physical Journal E*, 34:1–24, 2011.

[RCH+12] Peter Ronhovde, Saurish Chakrabarty, Dandan Hu, M Sahu, Kisor K Sahu, Kenneth F Kelton, Nicholas A Mauro, and Zohar Nussinov. Detection of hidden structures for arbitrary scales in complex physical systems. *Scientific reports*, 2(1):329, 2012.

[RK17] C. Patrick Royall and Walter Kob. Locally favoured structures and dynamic length scales in a simple glass-former. *Journal of Statistical Mechanics: Theory and Experiment*, 2017(2), 2017. arXiv: 1611.03314 Publisher: IOP Publishing.

[RPK17] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics Informed Deep Learning (Part I): Data-driven Solutions of Nonlinear Partial Differential Equations. *arXiv:1711.10561 [cs, math, stat]*, November 2017. arXiv: 1711.10561.

[RPS+20] D Richard, S Patinet, E Stanifer, B Shang, SA Ridout, B Xu, G Zhang, PK Morse, J-L Barrat, et al. Predicting plasticity in disordered solids from structural indicators. *Physical Review Materials*, 4(11):113609, 2020.

[RTS20] C Patrick Royall, Francesco Turci, and Thomas Speck. Dynamical phase transitions and their relation to structural and thermodynamic aspects of glass physics. *The Journal of Chemical Physics*, 153(9), 2020.

[SCKL16] Samuel S. Schoenholz, Ekin D. Cubuk, Efthimios Kaxiras, and Andrea J. Liu. The Relationship Between Local Structure and Relaxation in Out-of-Equilibrium Glassy Systems. *Proceedings of the National Academy of Sciences*, 114(2):263–267, January 2016. arXiv: 1607.06969.

[SCS+16] Samuel S Schoenholz, Ekin D Cubuk, Daniel M Sussman, Efthimios Kaxiras, and Andrea J Liu. A structural approach to relaxation in glassy liquids. *Nature Physics*, 12(5):469–471, 2016.

[SGB22] Camille Scalliet, Benjamin Guiselin, and Ludovic Berthier. Thirty milliseconds in the life of a supercooled liquid. July 2022. arXiv:2207.00491 [cond-mat].

[SGP+22] Hannes Stärk, Octavian Ganea, Lagnajit Pattanaik, Dr.Regina Barzilay, and Tommi Jaakkola. EquiBind: Geometric deep learning for drug binding structure

prediction. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 20503–20521. PMLR, 17–23 Jul 2022.

[SGPC+23] Christoph Schür, Lilian Gasser, Fernando Perez-Cruz, Kristin Schirmer, and Marco Baity-Jesi. A benchmark dataset for machine learning in ecotoxicology. *Scientific Data*, 10(1):718, 2023.

[SHD+24] Arne Schneuing, Charles Harris, Yuanqi Du, Kieran Didi, Arian Jamasb, Ilia Igashov, Weitao Du, Carla Gomes, Tom Blundell, Pietro Lio, Max Welling, Michael Bronstein, and Bruno Correia. Structure-based drug design with equivariant diffusion models, 2024.

[Shi18] T. Shinzato. Validation of the replica trick for simple models. *Journal of Statistical Mechanics: Theory and Experiment*, 2018.

[SHSS22] Hayato Shiba, Masatoshi Hanai, Toyotaro Suzumura, and Takashi Shimokawabe. Predicting the entire glassy dynamics from static structure by machine learning relative motion. September 2022. arXiv:2206.14024 [cond-mat].

[SHSS23] Hayato Shiba, Masatoshi Hanai, Toyotaro Suzumura, and Takashi Shimokawabe. Botan: Bond targeting network for prediction of slow glassy dynamics by machine learning relative motion. *The Journal of Chemical Physics*, 158(8), 2023.

[SHW22] Victor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) Equivariant Graph Neural Networks. *arXiv:2102.09844 [cs, stat]*, February 2022. arXiv: 2102.09844.

[SLRR14] S. S. Schoenholz, A. J. Liu, R. A. Riggleman, and J. Rottler. Understanding plastic deformation in thermal glasses from single-soft-spot dynamics. *Physical Review X*, 4(3):1–11, 2014. arXiv: 1404.1403.

[SSCL17] Daniel M. Sussman, Samuel S. Schoenholz, Ekin D. Cubuk, and Andrea J. Liu. Disconnecting structure and dynamics in glassy thin films. *Proceedings of the National Academy of Sciences*, 114(40):10601–10605, October 2017.

[SST92] H. S. Seung, H. Sompolinsky, and N. Tishby. Statistical mechanics of learning from examples. *Physical Review A*, 45(8):6056–6091, April 1992.

[STC+18] Tristan A Sharp, Spencer L Thomas, Ekin D Cubuk, Samuel S Schoenholz, David J Srolovitz, and Andrea J Liu. Machine learning determination of atomic dynamics at grain boundaries. *Proceedings of the National Academy of Sciences*, 115(43):10943–10947, 2018.

[SW82] Frank H. Stillinger and Thomas A. Weber. Hidden structure in liquids. *Physical Review A*, 25(2):978–989, February 1982.

[SW24] Antonio Sclocchi and Matthieu Wyart. On the different regimes of Stochastic Gradient Descent. *Proceedings of the National Academy of Sciences*, 121(9):e2316301121, February 2024. arXiv:2309.10688 [cond-mat, stat].

[TGB+23a] Artur P. Toshev, Gianluca Galletti, Johannes Brandstetter, Stefan Adami, and Nikolaus A. Adams. E(3) equivariant graph neural networks for particle-based fluid mechanics, 2023.

[TGB+23b] Artur P. Toshev, Gianluca Galletti, Johannes Brandstetter, Stefan Adami, and Nikolaus A. Adams. Learning lagrangian fluid mechanics with e(3)-equivariant graph neural networks, 2023.

[THZ20] Kaihua Tang, Jianqiang Huang, and Hanwang Zhang. Long-Tailed Classification by Keeping the Good and Removing the Bad Momentum Causal Effect. 2020.

[TKVB22] Christos Thrampoulidis, Ganesh R Kini, Vala Vakilian, and Tina Behnia. Imbalance Trouble: Revisiting Neural-Collapse Geometry. 2022.

[TRL22] Indrajit Tah, Sean A Ridout, and Andrea J Liu. Fragility in glassy liquids: A structural approach based on machine learning. *The Journal of Chemical Physics*, 157(12), 2022.

[TSK+18] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation- and translation-equivariant neural networks for 3D point clouds. *arXiv (unpublished)*, May 2018. arXiv: 1802.08219.

[TT19] Hua Tong and Hajime Tanaka. Structural order as a genuine control parameter of dynamics in simple glass formers. *Nature Communications*, 10(1):5596, December 2019.

[TTR17] Francesco Turci, Gilles Tarjus, and C. Patrick Royall. From Glass Formation to Icosahedral Ordering by Curving Three-Dimensional Space. *Physical Review Letters*, 118(21):1–5, 2017. arXiv: 1609.03044.

[vBKvS90] B. W. H. van Beest, G. J. Kramer, and R. A. van Santen. Force fields for silicas and aluminophosphates based on ab initio calculations. *Physical Review Letters*, 64:1955–1958, 1990.

[Vee24] Lars Veefkind. A probabilistic approach to learning the degree of equivariance in steerable cnns. Master's thesis, Universiteit Van Amsterdam, August 2024. Available at `https://uvagedl.github.io/Learning_the_Degree_of_Equivariance_for_Steerable_CNNs_fixed_typo_compressed.pdf`.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[VZ85] JJM Verbaarschot and MR Zirnbauer. Critique of the replica trick. *Journal of Physics A: Mathematical and General*, 18(7):1093–1110, 1985.

[WCHF04] Asaph Widmer-Cooper, Peter Harrowell, and H. Fynewever. How Reproducible Are Dynamic Heterogeneities in a Supercooled Liquid? *Physical Review Letters*, 93(13):135701, September 2004.

[WGW+18] Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco S Cohen. 3d steerable cnns: Learning rotationally equivariant features in volumetric data. *Advances in Neural Information Processing Systems*, 31, 2018.

[WRB93] Timothy L. H. Watkin, Albrecht Rau, and Michael Biehl. The statistical mechanics of learning a rule. *Reviews of Modern Physics*, 65(2):499–556, April 1993.

[WYKN20] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, 53(3):1–34, 2020.

[WZE17] Lei Wu, Zhanxing Zhu, and Weinan E. Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes, November 2017. arXiv:1706.10239 [cs].

[XM89] Yu Xie and Charles F. Manski. The Logit Model and Response-Based Samples. *Sociological Methods & Research*, 17(3):283–302, February 1989.

[YTWM00] Kenji Yamanishi, Jun-Ichi Takeuchi, Graham Williams, and Peter Milne. On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 320–324, 2000.

[Zde20] Lenka Zdeborová. Understanding deep learning is also a job for physicists. *Nature Physics*, 16(6):602–604, June 2020.

[ZG24a] D Zhemchuzhnikov and S Grudinin. Ilpo-net: Network for the invariant recognition of arbitrary volumetric patterns in 3d. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 319–332. Springer, 2024.

[ZG24b] D Zhemchuzhnikov and S Grudinin. On the fourier analysis in the so(3) space: Equilopo network. *arXiv preprint arXiv:2404.15979*, 2024.

[ZM03] Jianping Zhang and Inderjeet Mani. kNN Approach to Unbalanced Data Distributions: A Case Study involving Information Extraction. 2003.

[ZSZ+19] Lizong Zhang, Xiang Shen, Fengming Zhang, Minghui Ren, Binbin Ge, and Bo Li. Anomaly detection for power grid based on time series model. In *2019 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, pages 188–192. IEEE, 2019.

[ZXY+22] Ge Zhang, Hongyi Xiao, Entao Yang, Robert JS Ivancic, Sean A Ridout, Robert A Riggleman, Douglas J Durian, and Andrea J Liu. Structuro-elasto-plasticity model for large deformation of disordered solids. *Physical Review Research*, 4(4):043026, 2022.