THE UNIVERSITY OF CHICAGO

ON LEARNING AND OPTIMIZATION IN INVERSE PROBLEMS WITH GROUP STRUCTURED LATENT VARIABLES

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE DIVISION OF THE PHYSICAL SCIENCES
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

DEPARTMENT OF STATISTICS

BY

SOUNAK PAUL

CHICAGO, ILLINOIS

AUGUST 2024

To my family.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

The completion of this PhD thesis marks the culmination of an incredible journey that would not have been possible without the support and encouragement of many individuals and institutions. I am profoundly grateful to all those who have contributed to my academic and personal growth.

First, I would like to express my deepest gratitude to my supervisor Yuehaw Khoo, for his unwavering support, guidance, and patience. Your expertise, insightful feedback, and encouraging words have been instrumental in shaping this research and helping me navigate the complexities of my study. I am also thankful to the rest of the members of my thesis committee, Nir Sharon and Claire Donnat, for their valuable time and constructive criticism. Your diverse perspectives and suggestions have significantly enhanced the quality of my work. All of you have left an indelible mark on my academic and professional life.

I extend my heartfelt thanks to my colleagues and friends at UChicago Statistics, whose camaraderie and intellectual discussions have enriched my research experience. Special thanks to my department for providing the financial support necessary to conduct my research. Your funding was crucial in allowing me to pursue this academic endeavor without financial worries.

I am immensely grateful to my entire family, especially my parents and sister, for their unconditional love, patience, and encouragement. Your belief in my abilities has been a constant source of motivation during turbulent times. To my friends, both near and far, thank you for your understanding, encouragement, and for providing much-needed breaks from my studies. Your friendship has been invaluable in my journey.

To everyone who has supported me in any way, thanks a lot. This achievement is a testament to your contributions and belief in my potential.

ABSTRACT

Inverse problems are ubiquitous in science and engineering, manifesting whenever we seek to determine the underlying causes or parameters that give rise to observed data. These problems often involve latent variables, which in many cases, follow a group structure. In this class of inverse problems, we aim to estimate an unknown function after being distorted by a group action and observed via a known operator, with the observations typically being contaminated with a non-trivial level of noise. Two particular such problems of interest in this thesis are multireference alignment (MRA) and single-particle reconstruction (SPR) in cryo-electron microscopy (cryo-EM). SPR is a widely used technique for estimating the 3-D volume of a single macromolecule (often referred to as *volume* or *signal*) given several of its noisy 2-D projections taken at unknown viewing angles. In Chapter 1 we discuss the problem setting and mathematically formulate both MRA and cryo-EM.

The method of moments (MoM) is a powerful technique used to suppress the noise, and provide a low-resolution *ab initio* initialization for the 3-D structure in cryo-EM. Maximum likelihood estimation (MLE) based approaches like Expectation Maximization (EM) or Empirical Risk Minimization (ERM) are widely used for iterative refinement of the *ab initio* structure to obtain high-resolution reconstructions. This thesis broadly deals with developing deep neural networks for solving inverse problems with group structured latent variables via MoM, and accelerating MLE-based methods using variance reduction techniques and second-order information.

In Chapter 2 we suggest using the method of moments approach for both problems while introducing deep neural network priors. In particular, given a set of datasets, each containing observations corresponding to a single signal and distribution, our neural networks should output the signals and the distribution of group elements, with moment pairs of each dataset being the input. For MRA, we demonstrate the advantage of using the trained network to accelerate the convergence of the reconstruction of signals from moments coming from

an unknown dataset. Finally, we use our method to reconstruct simulated and biological volumes in the cryo-EM setting.

Chapter 3 is a direct extension of Chapter 2, in which we introduce MoM-net, a deep neural network for learning the moment inversion map for a more generalized cryo-EM setting where we assume the presence of small shifts in the projections. Our neural network is trained to output the spherical harmonic coefficients of the volumes along the distribution of rotations and shift variance, with moments from a set of datasets being the input. We also demonstrate the acceleration of convergence for the reconstruction using the trained neural network in this general cryo-EM setting, and use our method to reconstruct biological volumes.

In Chapter 4 we study the same problems but using a different framework, i.e. maximum likelihood. Maximization of the likelihood function is usually carried out using first-order ERM and EM methods which suffer from slow convergence rates, while their stochastic versions have high variance in parameter updates. Stochastic variance-reduced gradient (SVRG) methods have been proposed in the literature to improve convergence rates and stability by reducing the variance of the stochastic updates. This chapter thus explores the application of SVRG and stochastic variance-reduced EM (sEM-vr) methods, along with their second-order accelerated variants, in solving MRA and SPR. A second-order acceleration of sEM-vr is also proposed. We conduct extensive experiments on simulated datasets illustrating the applicability of variance-reduced methods for both of these problems.

We end with Chapter 5, where we provide final thoughts on the overarching theme of this thesis, and discuss the strengths and drawbacks of our methods, along with potential future research steps.

CHAPTER 1

INTRODUCTION

1.1 Overview

Inverse problems are a significant area of study in mathematics and applied sciences, characterized by their fundamental role in various fields such as physics [47, 58], engineering [39, 25], medical imaging [72, 12], and geophysics [62, 49]. The core idea in these problems is to infer certain unknown parameters or inputs of a system from observable outputs. This contrasts with direct problems, where the outputs are determined given specific inputs and system parameters.

Inverse problems often involve latent variables, which significantly increase its complexity. Mathematically, a direct problem involving latent variables can be described by

$$y_j = \mathcal{F}(x, h_j), \qquad j = 1, \dots, N, \tag{1.1}$$

where $\mathcal{F}$ is a known forward operator mapping input x and latent variables $h_1, \dots, h_N$ to observed data $y_1, \dots, y_N$. Then the inverse problem involves finding x (and $h_1, \dots, h_N$) given $y_1, \dots, y_N$.

A particular class of inverse problems of interest involve incorporating the effect of a group on a data model, i.e. the associated latent variables follow a group structure. The resulting solution is determined up to an arbitrary group action, meaning that the solutions form an *orbit*. This class of estimation problems is referred to as **orbit recovery problems** [22, 6], and are crucial in various fields of science and engineering, ranging from signal processing to structural biology. For instance, medical tomography often collects imaging data that undergoes unknown transformations. Along with pixel-wise noise, each image may experience rotation, translation, flipping, or other group actions in an unknown manner. This thesis shall examine two problems in this category and develop new approaches to solving them,

while also proposing methods to accelerate existing approaches.

1.2 Notation

$\circ$	Function composition.
$\odot$	Hadamard (also known as element-wise) product of vectors/matrices.
$\otimes$	Tensor product.
$\hat{v}$	Estimator of v.
$\widehat{v}$	Fourier transformation of v.
$\mathcal{I}$	Unit interval $\left[-\frac{1}{2}, \frac{1}{2}\right]$.
X_1	n equispaced points on $\mathcal{I}$.
K_1	n equispaced points on $[-\pi, \pi]$.
X_2	n^2 equispaced points on $\mathcal{I}^2$.
K_2	n^2 equispaced points on $[-\pi, \pi]^2$.

1.3 Orbit recovery problems

We first begin by formulating the general problem of orbit recovery. Let v be an unknown scalar-valued object defined as a function

$$v \colon \Omega \to \mathbb{R}, \tag{1.2}$$

and let G be a group with a well-defined action on v, that is $G \curvearrowright \Omega$. One class of estimation problems we are concerned with consists of the following general formulation. Our goal is to estimate the function v (which we shall refer to as the *signal*) from N observed samples,

$$v_j = \mathcal{A}(g_j \circ v) + \varepsilon_j, \quad g_j \sim \rho, \quad j = 1, \dots, N, \tag{1.3}$$

where $\left\{\varepsilon_j\right\}_{j=1}^N$ is a set of i.i.d. random noise terms, $\mathcal{A}$ is a known operator, and $\{g_j\}_{j=1}^N$ is a set of i.i.d. random group elements distributed according to some distribution ρ on G. These are treated as latent variables or nuisance parameters for our problem since the objective is to estimate v. Note that one can only estimate v up to a group action, since for any estimator $\hat{v}$ and $\{\hat{g}_j\}_{j=1}^N$ for the object and latent group elements respectively, $g \circ \hat{v}$ and $\{\hat{g}_j g^{-1}\}_{j=1}^N$ give another set of equivalent estimators for any fixed $g \in G$. Hence our goal becomes the *orbit recovery* of v.

In many orbit recovery problems, $\mathcal{A}$ is linear. In that case, customarily, in the lower-noise regime where the magnitude of ε_j is smaller than that of v, a solution to (1.3) is obtained using the following scheme. First $g_{ij} \approx g_i g_j^{-1}$ is estimated from v_i and v_j. The estimation procedure depends on the specific problem (see [70, 60]). Then one recovers the group elements $\{g_j\}_{j=1}^N$ from the set of their ratios $\{g_i g_j^{-1}\}_{i,j=1}^N$, i.e. solving a *synchronization problem* over G, after fixing g_1 to be the identity element. Then with a good estimation for $\{g_j\}_{j=1}^N$, we solve for v in problem (1.3) via solving a system of linear equations [3].

As the level of noise in the observations increases, the random noise heavily influences the alignment results so that even if the ground truth v is provided, finding the group action corresponding to any particular observation would be extremely error-prone. Thus, the assignment of group actions to the observations would incur large errors [68, 69]. A different approach consists of treating the group elements $\{g_j\}_{j=1}^N$ as nuisance parameters and having the signal be the primary estimation target. In other words, when considering a high level of noise, we focus on methods that marginalize over the nuisance parameters by treating them as random variables [11]. The estimation of v can be done via maximizing the marginalized posterior distribution that has v being the random variable or using a method of moments with moments formed by averaging v_j's such that there is no dependency on g_j's.

In the next two sections, we are going mathematically formulate two orbit recovery problems, Multireference Alignment (MRA) and single particle reconstruction (SPR) in Cryogenic

Electron Microscopy (cryo-EM). MRA involves estimating a signal from the observation of noisy, circularly shifted copies of it. This model, which has its origins in both signal processing [81] and structural biology [64, 73], provides a foundation for exploring the relationship between the group structure, noise levels, and the possibility of recovery [74, 1, 43]. The second problem, i.e. SPR, deals with 3D volume reconstruction in cryo-EM, as discussed in [66]. The goal is to retrieve a 3D volume from 2D noisy images that result from rotating the volume and applying a fixed tomographic projection. The dataset is a set of 2D images, which are usually heavily contaminated with noise.

1.4 Multireference Alignment (MRA)

Multireference alignment is a critical problem in computational science that arises in various questions across science and engineering, like signal processing [81, 26], image recognition [57, 23] and robotics [59]. This problem, along with its variant of windowed MRA (explored in Chapter 4), serves as a simplification for more complex ones that feature repeated observations of a signal subject to latent group actions and additive measurement noise, like SPR. MRA involves aligning multiple noisy observations of a signal or an object that have been transformed by unknown translations. The objective is to recover the underlying structure or signal that is common to all the observations. The optimization landscape is often non-convex, leading to multiple local minima that can trap optimization algorithms. High dimensionality and low signal-to-noise ratio (SNR) are two other challenges.

For the MRA model, $\mathcal{A}$ from (1.3) is the identity. In this situation, the unknown signal v is defined on a unit, symmetric segment $\mathcal{I} = [-\frac{1}{2}, \frac{1}{2}]$. Namely, the signal is $v : \mathcal{I} \to \mathbb{R}$, and we further assume it is a periodic, band-limited function. Let G be the group of circular translations (rotations) on $\mathcal{I}$, whose elements s_j shift v in the following manner,

$$s_j \circ v := v(\cdot - s_j). \tag{1.4}$$

Here, we interpret the difference as modulo the segment, namely $\cdot - s_j$ is always in $\mathcal{I}$. The data, i.e. observations, we obtain are of the form

$$v_j = s_j \circ v(X_1) + \epsilon_j, \quad j = 1, \ldots, N \tag{1.5}$$

where $\epsilon_j \sim N(0, \sigma^2 I_n)$, and X_1 is a set of n equispaced points on $\mathcal{I}$. Some sample MRA observations corresponding to a Haar-like signal with different noise levels, are displayed in Figure 1.1.

We next formulate the MRA problem in the Fourier domain. For convenience, we discuss the case when there is no noise. Let $\widehat{v}_j$ be the Fourier transform of v_j, in this case, a shift s_j becomes a phase, i.e.

$$\widehat{v}_j(k) = \exp(iks_j)\widehat{v}(k), \ k \in [-\pi, \pi]. \tag{1.6}$$

The frequency k has a natural bandlimit $|k| \leq \pi$ since the signal v_j is usually provided on n discretized points in $\mathcal{I}$, where n is chosen to satisfy its Nyquist frequency. As for our observation, let K_1 be the set of n equispaced points between $[-\pi, \pi]$. Then, we have

$$\widehat{v}_j(k) = \exp(iks_j)\widehat{v}(k), \quad k \in K_1. \tag{1.7}$$

Henceforth, for brevity, we use $\widehat{v}_j(K_1) = \exp(iK_1 s_j) \odot \widehat{v}(K_1)$ instead of the pointwise notation, where "$\odot$" denotes the Hadamard product (see Section 1.2). Hence in presence of noise, our observations become

$$\widehat{v}_j(K_1) = \exp(iK_1 s_j) \odot \widehat{v}(K_1) \ + \ \widehat{\epsilon}_j, \quad k \in K_1, \tag{1.8}$$

where $\widehat{\epsilon}_j$ is the Fourier transform of ϵ_j from (1.5).

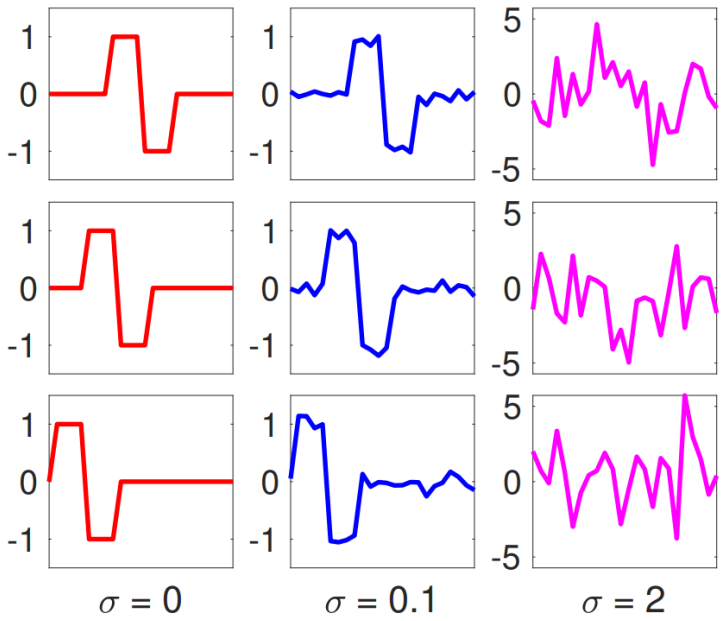

Figure 1.1: MRA observations as per (1.5). The left column presents three observations corresponding to different shift in zero-noise case. The middle and right columns correspond to the same translations but with low and high noise levels respectively. Image taken from [1].

1.5 Cryogenic electron microscopy (cryo-EM)

Cryogenic electron microscopy (cryo-EM) has revolutionized the field of structural biology by allowing researchers to visualize macromolecules at near-atomic resolution. Unlike traditional methods such as X-ray crystallography or nuclear magnetic resonance (NMR) spectroscopy, cryo-EM does not require the crystallization of the sample or the use of large quantities of the material. Among the various techniques within cryo-EM, single particle reconstruction (SPR) [7, 71, 52] has become particularly significant. This technique involves the analysis of individual particles suspended in a thin ($\sim$ 100 nm) layer of vitreous ice, capturing their images in numerous orientations and subsequently reconstructing their 3D structure from these 2D projections [24, 50, 45]. Figure 1.2 shows a simplified diagram of the acquisition of 2D projections from 3D biomolecular volumes.

The flash-freezing of biological molecules is done in a way that preserves their native state. The sample, which contains the particles of interest, is vitrified to avoid the formation of ice

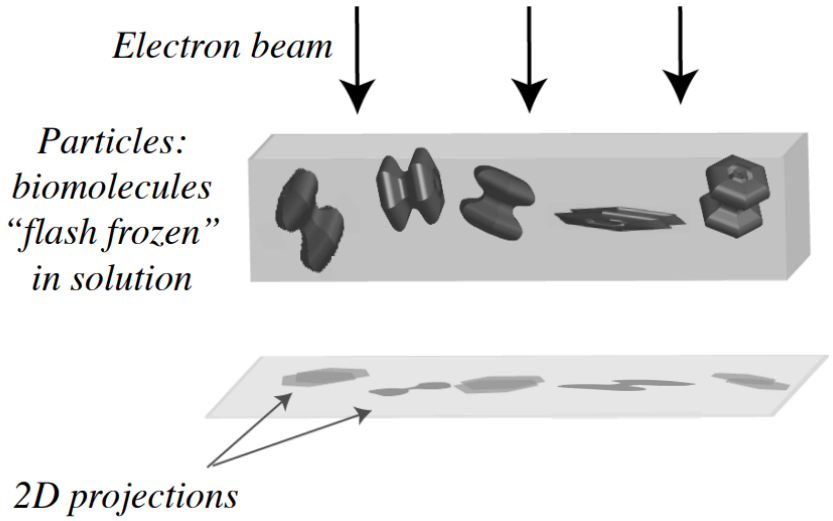

Figure 1.2: Acquisition of 2D projections from 3D biomolecules. Image taken from [18].

crystals that could damage the structures. These vitrified samples are then imaged using an electron microscope, producing a series of 2D images, each representing a projection of the particles in random orientations. Figure 1.3 depicts a single cryo-EM image corresponding to a random rotation of a single volume.

The challenge of single particle reconstruction lies in transforming these 2D images into a coherent 3D model. This transformation is achieved through a series of computational steps. First, individual particle images are extracted from the micrographs. These images are typically noisy and low in contrast, requiring sophisticated algorithms for alignment and averaging. The next step involves determining the relative orientations of the particles. Once the orientations are known, the images are combined to reconstruct the 3D structure using techniques such as weighted back-projection or iterative refinement methods.

The computational aspects of single particle reconstruction are highly demanding. One of the primary challenges is dealing with the signal-to-noise ratio. Cryo-EM images are often dominated by noise due to the low dose of electrons used to avoid radiation damage to the sample. As a result, distinguishing between signal (the true structure of the particle) and

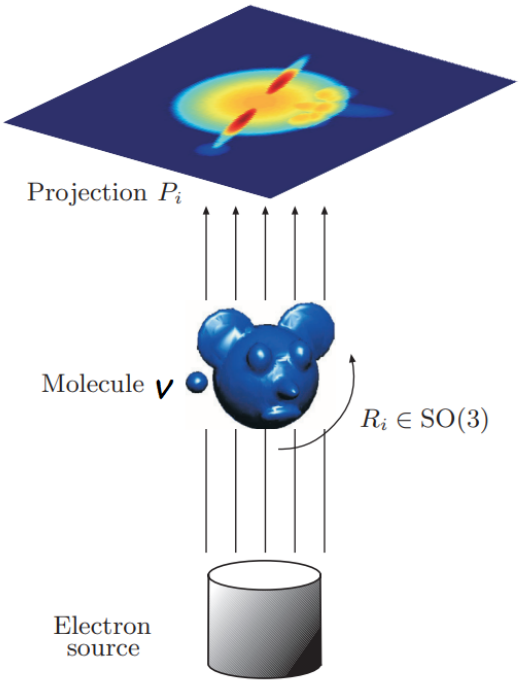

Figure 1.3: Schematic drawing of the cryo-EM imaging process. Image taken from [70].

noise becomes a crucial task. Noise reduction techniques and averaging multiple particle images help in enhancing the signal, but these processes require substantial computational power and advanced algorithms. Another major challenge is the determination of particle orientations. Since the particles are randomly oriented in the ice, accurately determining these orientations is essential for successful reconstruction. Traditional methods involve exhaustive searching and matching, which are computationally expensive. Another approach is based on the method of moments (MoM) that exploits the known analytical relation between the moments of the data with those of the 3D volume, but this only leads to a low-dimensional *ab-initio* reconstruction (more in Chapters 2 and 3). Modern approaches employ machine learning and optimization techniques to improve the accuracy and efficiency of orientation determination [40, 78, 79, 63, 33].

The field of cryo-EM has seen remarkable advancements in both hardware and software, significantly improving the resolution and efficiency of single particle reconstructions. One

of the key hardware developments is the advent of direct electron detectors [76], which offer superior sensitivity and faster readout speeds compared to traditional cameras, resulting in higher quality images with better signal-to-noise ratios. On the software side, numerous tools and algorithms have been developed to address the challenges of single particle reconstruction. Programs such as RELION [44, 63, 80], cryoSPARC [56], and cisTEM [27] provide comprehensive pipelines for processing cryo-EM data, from initial image preprocessing to final 3D reconstruction. These tools incorporate advanced image processing techniques, including maximum likelihood estimation, Bayesian inference, and deep learning, to enhance the accuracy and resolution of reconstructions.

The problem (1.3) also serves as a simplified model of single-particle cryo-EM, where the operator $\mathcal{A}$ is a tomographic projection along a fixed axis. Let us denote by $v\colon \mathbb{R}^3 \to \mathbb{R}$ the Coulomb potential of the 3D volume we aim to determine, where we assume that v is compactly supported in a ball of radius $\frac{1}{2}$ around the origin, that is inside $\mathcal{I}^3$. We define the composition of R_j with the volume v as

$$R_j \circ v \ (x,y,z) \ = \ v\left(R_j^T [x \ y \ z]^T\right), \quad (x,y,z) \in \mathcal{I}^3, \tag{1.9}$$

viewing R_j as a 3×3 matrix in the right hand side of (1.9) since $\mathrm{SO}(3) \subset \mathbb{R}^{3\times 3}$. Let $\mathcal{P}\colon \mathbb{R}^3 \to \mathbb{R}^2$ be the operator that projects a 3D volume along the z axis to a 2D image, i.e.

$$\mathcal{P} \circ v \ (x,y) \ = \ \int_{-\infty}^{\infty} v\,(x,y,z)\,dz, \quad (x,y,z) \in \mathcal{I}^2. \tag{1.10}$$

Then, a standard image formation model in the absence of noise, after filtering the effect of the contrast transfer function (CTF), image cropping, and centering, is (see [24, 29]),

$$v_j = \mathcal{P} \circ R_j \circ v, \quad j = 1, \ldots, N, \tag{1.11}$$

where $R_j \in \mathrm{SO}(3)$ are the unknown group elements. In practice, the real noisy observations that are provided to us are of the form

$$v_j = \mathcal{P} \circ R_j \circ v(X_2) \; + \; \epsilon_j, \quad j = 1, \ldots, N \tag{1.12}$$

where $\epsilon_j \sim N(0, \sigma^2 I_{n^2})$ and X_2 is n^2 equispaced points on $\mathcal{I}^2$.

To avoid the computationally intensive integration in (1.10), we reformulate our problem in the Fourier domain. There, we can exploit the Fourier Slice Theorem to speed up computation significantly. We define $\widehat{v}\colon [-\pi, \pi]^3 \to \mathbb{C}$ as the Fourier transform of v, and $S\colon [-\pi, \pi]^2 \to \mathbb{C}$ as the slice operator given as

$$S \circ \widehat{v}(k_x, k_y) \; = \; \widehat{v}\left(k_x, k_y, 0\right), \tag{1.13}$$

i.e., $S \circ \widehat{v}$ is obtained by slicing $\widehat{v}$ across the plane given by $z = 0$. Then, the Fourier Slice Theorem [46] states that:

$$\mathcal{F}_{2D} \circ \mathcal{P} \circ R \; = \; \mathcal{S} \circ R \circ \mathcal{F}_{3D}, \tag{1.14}$$

where $R \in \mathrm{SO}(3)$, $\mathcal{F}_{2D}$ and $\mathcal{F}_{3D}$ are the 2D and 3D Fourier transformations, respectively. Therefore, in the no-noise setting, the equivalent of (1.11) becomes,

$$\widehat{v}_j(k_x, k_y) \; = \; S \circ R_j \circ \widehat{v}(k_x, k_y), \quad (k_x, k_y) \in [-\pi, \pi]^2, \tag{1.15}$$

where $\widehat{v}_j$ is the Fourier transform of v_j. Let K_2 be a grid of n^2 equispaced points on $[-\pi, \pi]^2$, flattened as a one-dimensional vector. Now, our noisy observations are

$$\widehat{v}_j(K_2) \; = \; S \circ R_j \circ \widehat{v}(K_2) \; + \; \widehat{\epsilon}_j, \tag{1.16}$$

with $\widehat{\epsilon}_j$ being the Fourier transform of ϵ_j from (1.11).

The field of cryo-EM continues to evolve rapidly, driven by ongoing advancements in technology and methodology. Despite the significant computational and technical challenges, advances in hardware and software have enabled researchers to achieve unprecedented levels of detail and accuracy. The impact of these developments is evident in the numerous high-resolution structures obtained and the insights they provide into fundamental biological processes. Future directions include the development of even more sensitive detectors, faster and more accurate computational algorithms, and methods for improving sample preparation and preservation. Additionally, integrating cryo-EM with other structural biology techniques, such as cryo-electron tomography (cryo-ET) [41] and single-molecule fluorescence microscopy [67], holds promise for providing a more comprehensive understanding of cellular structures and dynamics. As technology continues to advance, the potential for cryo-EM to transform our understanding of molecular biology remains immense, heralding a new era of structural biology.

1.6 Outline of thesis

A brief outline of the thesis is as follows. Chapters 2 and 3 deal with developing deep neural network priors for solving orbit recovery problems via MoM. In Chapter 2, given a set of MRA datasets, each containing observations corresponding to a single signal and distribution of shifts, our neural networks are trained to output the signals and the distribution of group elements, with moment pairs of each dataset being the input. We then demonstrate the advantage of using the trained neural network to accelerate the convergence for the reconstruction of signals (and distribution of shifts) from moments coming from a new, unknown dataset. Finally, we modify our method to reconstruct simulated and biological volumes in the cryo-EM setting. This chapter is adapted from the author's paper [32], which is joint work with Yuehaw Khoo and Nir Sharon.

Motivated by the promise shown by our architecture in Chapter 2, we develop a slightly different framework in Chapter 3, and call it MoM-net, a deep neural network for learning the moment inversion map for the SPR problem. The setting is a lot more general than in Chapter 2, and here we assume the presence of small shifts in the projections, i.e. the projection of the unknown volume is not perfectly centered. Our neural network is trained in a supervised way with a new loss function, so that provided with moments from a set of datasets as input, it can output the spherical harmonic coefficients of the volumes along the distribution of rotations and shift variance. As in Chapter 2, each dataset during training corresponds to a single volume, distribution of rotations, and shift variance. We also demonstrate a slight acceleration of convergence for the reconstruction using the trained neural network in this general cryo-EM setting. Finally we illustrate the superiority of MoM-net in the recovery of volumes in presence of shifts compared to our previous framework, and use our method to reconstruct biological volumes.

In the last chapter of this thesis, i.e. Chapter 4, we shift our attention from MoM to the MLE-based methods. Unlike the framework of the previous chapters where we only consider the first and second moments for reconstruction, the likelihood function encapsulates information from moments of all orders, and hence provides high-resolution reconstructions. The likelihood function is maximized using first-order ERM and EM methods when solving orbit recovery problems, which come with challenges in the form of slow convergence rates and multiple passes over the entire dataset. Stochastic versions of these methods mitigate the second challenge but the convergence rate decreases further owing to high variance in the calculated parameter updates. In the literature, stochastic variance-reduced gradient (SVRG) methods have been proposed to mitigate these issues, and they display improved convergence rates and stability by reducing the variance of stochastic gradients. Thus, we explore the application of SVRG and stochastic variance-reduced EM (sEM-vr) methods, along with their second-order accelerated variants, in solving orbit recovery problems, par-

ticularly MRA and SPR. A second-order acceleration of sEM-vr is also proposed, which is an original contribution. We also conduct extensive experiments for both these problems on simulated datasets, illustrating the applicability of variance-reduced methods and their second-order variants for orbit recovery.

In the final chapter of this thesis, namely Chapter 5, we describe the strengths and weaknesses of our main methods and provide final thoughts on the overarching theme of this thesis. We also discuss potential future steps of research along with some ideas on how to approach them.

CHAPTER 2

DEEP NEURAL NETWORK PRIOR FOR ORBIT RECOVERY FROM METHOD OF MOMENTS

2.1 Introduction

The method of moments (MoM) is a classical estimation technique that has been adapted in modern forms to provide a powerful computational tool for solving large-scale problems, especially when dealing with high noise levels. When used to solve orbit recovery problems, the MoM consists of two stages. First, given a dataset corresponding to a single ground truth signal and distribution of group elements, we compute the observable moments from the data by averaging the low-order statistics of any observation. The second stage involves retrieving the required signal from the observable moments by analyzing the relationship between the observable and analytical moments, applying moments-matching (described in Section 2.4.1), and deriving the unknown parameters from it. This second stage is the focus of this study.

The usage of MoM is advantageous in several ways. Its robustness is derived from the fact that noise is averaged out during the computation of observable moments. Namely, the effect of noise can be rendered insignificant given enough data, as described in Section 2.4.1. Furthermore, MoM gleans information about the data only through the moments, so it does not require multiple passes over the dataset. This is beneficial for dealing with huge datasets, as the moment calculation from the data takes place only in the first stage and in one pass [66, 69]. However, this method does have a major drawback. We can lose resolution since we are not using information from all the moments. MoM thus leads to low-dimensional (i.e. low-resolution) reconstructions. Fortunately, our focus is mainly to recover an *ab-initio* model; hence a low-dimensional reconstruction suffices. In the case of cryo-EM, this *ab-initio* model is used as an initialization for iterative refinement algorithms, where reconstruction

enables several possible conformations by further refinement [24].

This chapter introduces a new version of the method of moments that incorporates a neural network for tackling orbit recovery problems. In particular, we demonstrate the effectiveness of this approach for the two orbit recovery problems discussed earlier: Multireference Alignment (MRA) and single-particle Cryogenic Electron Microscopy (cryo-EM) modeling. Learning algorithms have recently taken a central role in cryo-EM computational methods: a deep neural network for modeling continuous heterogeneity (3DFlex) [55], ab initio neural reconstruction [78, 79, 40], and many other parts of the cryo-EM pipeline [30, 10, 33], to name a few. However, noise resilience remains one of the most significant challenges in cryo-EM 3D reconstruction. The proposed neural network-based method of moments technique provides a promising alternative that addresses this challenge effectively while also addressing the additional challenge of scalability.

In our method, we treat the group elements of each problem as random variables and consider them as nuisance parameters or latent variables. Rather than estimating them directly, we aim to target their density function along with the unknown signal. Our method of moments incorporates neural networks to approximate the signal and distribution of group elements to achieve this. We demonstrate that in the case of multireference alignment, a neural network can mimic existing algorithms for solving the inverse problem from the moments. Moreover, we propose that the moment inversion process can be significantly improved by using neural networks which were previously trained in a supervised manner on similar instances of the recovery problem. In other words, given a set of MRA datasets, each containing observations corresponding to a single signal and distribution of shifts, our neural network is trained in a supervised manner to output the signals and the distribution of group elements, with moment pairs of each dataset being the input. Then given a moment pair from a new dataset as input, the output of the trained neural network serves as a good initialization for further refinement during the reconstruction process. Since the reconstruction process

often gets stuck at spurious local minimas due to the ill-posed nature of the inverse problem, this *good* initialization provides regularization and biases the final output towards itself, effectively acting like a *prior*. We therefore refer to our method as a neural network-based prior, or a **neural network prior**. Our approach to the MRA problem serves as a proof-of-concept, and we extend these techniques to the case of cryo-EM.

The chapter is organized as follows. In Section 2.2, we present the method of moments approach for the MRA model and cryo-EM model individually as special cases of our class of estimation problems (1.3). Next, Section 2.3 introduces neural network priors for representing the volume and distribution of group elements for both models. Next, Section 2.4 illustrates the performance of our neural network priors in the reconstruction of various simulated as well as real-world biological volumes. Finally, we conclude with Section 2.5, including a summary of the next steps in this line of research.

2.2 Method of moments for orbit recovery

The method of moments (MoM) is a classical technique to estimate parameters from observed statistics, and has already successfully been employed in multireference alignment (MRA) [1, 8] and cryo-EM recovery [66], where the operator $\mathcal{A}$ of (1.3) is either the identity or a tomographic projection, respectively. The group consists of circular shifts on MRA and 3D rotations in cryo-EM recovery. Then, the m-th moment is the expectation of the m-th-order tensor product of the samples with themselves, i.e., $v_j^{\otimes m}$ (see Section 1.2). Interestingly, the minimal number of moments to guarantee uniqueness also determines the sample complexity — the number of samples needed, as a function of noise level, in order to have a consistent estimation, see [1, 2, 52]. Therefore, when studying (1.3), the MoM plays a significant role as a baseline for designing computational algorithms and analyzing the sample complexity.

In the method of moments for multireference alignment, we define the analytic moments

as

$$M_F^1[\widehat{v},\rho](k_1) = \mathbb{E}_\rho\left(\frac{1}{N}\sum_{j=1}^N \widehat{v}_j(k_1)\right), \quad M_F^2[\widehat{v},\rho](k_1,k_2) = \mathbb{E}_\rho\left(\frac{1}{N}\sum_{j=1}^N \widehat{v}_j(k_1)\widehat{v}_j(k_2)^*\right). \tag{2.1}$$

Here, M_F^1 and M_F^2 are functions of $\hat{v}, \rho$. The goal is to retrieve $\widehat{v}$ from unbiased estimators $\hat{M}_F^1, \hat{M}_F^2$ of M_F^1, M_F^2 in the presence of noisy data via matching the moments. The procedure to obtain these moment estimators $\hat{M}_F^1, \hat{M}_F^2$ from the data, along with the moment-matching method, is described in Section 2.4.1.

Similarly for the cryo-EM model, the associated moments are

$$\begin{aligned} M_F^1[\widehat{v},\rho](k_x,k_y) &= \mathbb{E}_\rho\left(\frac{1}{N}\sum_{j=1}^N \widehat{v}_j(k_x,k_y)\right) \\ M_F^2[\widehat{v},\rho](k_x,k_y,k_x',k_y') &= \mathbb{E}_\rho\left(\frac{1}{N}\sum_{j=1}^N \widehat{v}_j(k_x,k_y)\widehat{v}_j(k_x',k_y')^*\right). \end{aligned} \tag{2.2}$$

We aim to retrieve $\widehat{v}$ by matching the moments $M_F^1[\hat{v},\rho](K_2,K_2)$ and $M_F^2[\hat{v},\rho](K_2,K_2)$ with some unbiased estimators $\hat{M}_F^1, \hat{M}_F^2$ in the presence of noisy data.

2.3 Neural network priors for method of moments

This section presents neural network (NN) approaches for reconstructing the signal v and distribution ρ in MRA and cryo-EM settings. The general strategy is to view both the signal and distribution as being mapped by a NN from the estimated moments $\hat{M}_F^1$ and $\hat{M}_F^2$, as various previous works have shown that for MRA, $\hat{M}_F^1, \hat{M}_F^2$ are generically sufficient statistics for estimating the signal and the distribution of shifts [1], while for cryo-EM, they have enough information for recovering a low-resolution reconstruction [66]. In the MRA case, we design an encoder that can map the empirical moments to discretized signal and

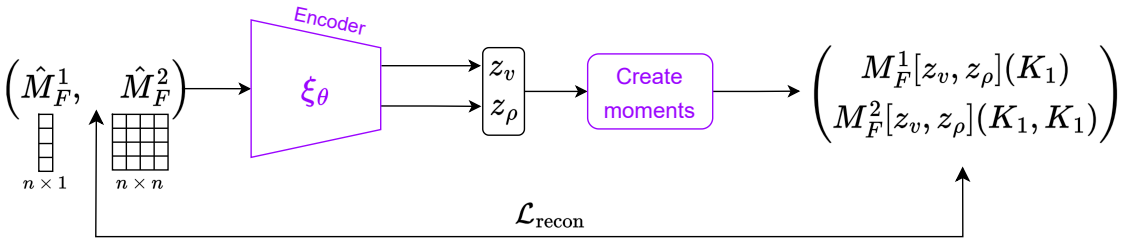

Figure 2.1: **Overview of our MRA pipeline**: The encoder ξ_θ takes moments $(\hat{M}_F^1, \hat{M}_F^2)$ as input, and outputs $z_\rho \in \mathbb{R}^n$, approximating a discretized probability density $\rho(X_1)$, and $z_v \in \mathbb{R}^n$ that approximates a discretized Fourier signal $\widehat{v}(K_1)$. Next, we use z_ρ and z_v to create $\left(M_F^1[z_v, z_\rho](K_1), M_F^2[z_v, z_\rho](K_1, K_1)\right)$ via equation (2.14), which we then compare with the inputs to the encoder, i.e., $(\hat{M}^1, \hat{M}^2)$ via the loss function $\mathcal{L}_{\text{recon}}$ (2.15).

density. In the cryo-EM case, we further design an encoder-decoder structure that allows us to take the moments as input and give a continuous representation of a 3D volume.

2.3.1 Neural networks for multireference alignment

In multireference alignment, we are given a set of datasets, each containing observations from the MRA model corresponding to a single underlying signal and distribution of shifts, while the signal and shift distribution differ over the set of datasets. We wish to train a neural network that can take the first two moments of a dataset as inputs, and output the underlying signal and density of shifts, which can further be used to initialize an iterative reconstruction algorithm. More precisely, we define $F \in \mathbb{C}^{n \times n}$ as the matrix representation of a normalized Fourier transform where $F^*F = I_n$, and X_1, K_1 as sets of n equispaced points on $\mathcal{I} = \left[-\frac{1}{2}, \frac{1}{2}\right]$ and $[-\pi, \pi]$ respectively. The main component is an encoder, i.e., a neural network ξ_θ, whose purpose is as follows. Let $\{\mathcal{D}_1, \ldots, \mathcal{D}_N\}$ be a set of N datasets corresponding to different pairs of signals and shift distributions, i.e. $\{(v_1, \rho_1), \ldots, (v_N, \rho_N)\}$. Then for any $j \in \{1, \ldots, N\}$, the neural network ξ_θ takes the empirical moments $(\hat{M}_F^1, \hat{M}_F^2)_j$ corresponding to dataset $\mathcal{D}_j$ as inputs, and outputs $(z_\rho, z_v)_j$, where $z_\rho \in \mathbb{R}^n$ approximates the discretized density $\rho_j(X_1)$ and $z_v \in \mathbb{R}^n$ approximates the discretized signal $\widehat{v}_j(K_1)$.

The encoder $\xi_\theta := (\xi_\theta^v, \xi_\theta^\rho)$ consists of two neural networks ξ_θ^v and ξ_θ^ρ, which are two 1D convolutional neural networks (CNNs) that take $\hat{M}_F^1 \in \mathbb{C}^n, \hat{M}_F^2 \in \mathbb{C}^{n\times n}$ as input vector fields supported on n grid points. Figure 2.1 provides an overview of our pipeline for MRA. While the details of the architectures are provided in Appendix A, here we provide motivations as to why a CNN has the capability to learn a mapping from the moments $\hat{M}_F^1 \in \mathbb{C}^n, \hat{M}_F^2 \in \mathbb{C}^{n\times n}$ to $\widehat{v}(K_1)$. For simplicity, suppose $|\widehat{v}(k)| = 1$. Using the definitions in (2.1) and the fact that translating v by s is equivalent to letting $\widehat{v}(k) \to \widehat{v}(k)\exp(iks)$, one can show that

$$M_F^2[\widehat{v}, \rho](k_1, k_2) = \widehat{v}(k_1)\widehat{\rho}(k_1 - k_2)\widehat{v}(k_2)^*, \tag{2.3}$$

as in [1]. In this case, $M_F^2[\widehat{v}, \rho](K_1, K_1)$ admits the eigendecomposition

$$\begin{aligned} M_F^2[\widehat{v}, \rho](K_1, K_1) &= \mathrm{diag}(\widehat{v}(K_1))F^*(F[\widehat{\rho}(k_1 - k_2)]_{k_1,k_2}F^*)F\mathrm{diag}(\widehat{v}(K_1)^*) \\ &= [\mathrm{diag}(\widehat{v}(K_1))]F^*\mathrm{diag}(\rho(X_1))[F\mathrm{diag}(\widehat{v}(K_1)^*)] \end{aligned} \tag{2.4}$$

since $[F\mathrm{diag}(\widehat{v}(K_1)^*)]$ is an orthogonal matrix (due to the assumption $|\widehat{v}(k)| = 1$). From this form, it is clear that the eigenvalues of $M_F^2[\hat{v}, \rho](K_1, K_1)$ are $\rho(X_1)$ and furthermore, the eigenvectors are $F\mathrm{diag}(\widehat{v}(K_1)^*)$. Since the spectral information of the second moments contains information concerning the signal and density, if a neural network can mimic a spectral method, then it can learn the mapping from moments to the signal and density.

The form of $M_F^2[\widehat{v}, \rho](K_1, K_1)$ in (2.4) suggests that it is a circulant matrix. Therefore if we want to devise a neural network that takes $\hat{M}_F^2 = M_F^2[\widehat{v}, \rho](K_1, K_1)$ (when there is no noise) as input and output the eigenvectors $F\mathrm{diag}(\widehat{v}(K_1)^*)$, we can use a neural network, composed of 1D convolutional layers, that takes $\hat{M}_F^2$ as a 1D n-dimensional vector field supported on n grid points. For example, to compute an eigenvector of $M_F^2[\widehat{v}, \rho](K_1, K_1)$, a

convolutional layer $l_1 : \mathbb{C}^n \to \mathbb{C}^n$ can take the form

$$l_1(u) = \frac{\hat{M}_F^2 u}{\|\hat{M}_F^2 u\|_2}. \tag{2.5}$$

One can think about $\hat{M}_F^2$ as the weights of the convolutional layer l_1, and the division by $\|\hat{M}_F^2 u\|_2$ as some nonlinearities in the NN. Repeated applications of l_1, gives an eigenvector of $M_F^2[\widehat{v}, \rho](K_1, K_1)$, due to the power method [5]. After obtaining an eigenvector, say for example $F(:,1)\widehat{v}(K_1(1))$ where $F(:,1)$ is the first column of F, the neural network can simply apply a layer of pointwise nonlinearities $l_2 : \mathbb{C}^n \to \mathbb{C}^n$ that performs

$$l_2(u(i)) = \frac{u(i)}{F(i,1)}, \quad i \in [n]. \tag{2.6}$$

Putting these elements together into a deep neural network, i.e., $l_2 \circ l_1 \circ \cdots \circ l_1$ should give $\widehat{v}(K_1(1))$. Similar operations can be carried out for other eigenvectors. We also use a similar structure for ξ_θ^ρ to output z_ρ that approximates $\rho(X_1)$, since it is clear that if $u = l_2 \circ l_1 \circ \cdots \circ l_1(\hat{M}_F^2)$ is an eigenvector of $\hat{M}_F^2$, applying another nonlinearity of the form

$$l_3(u) = \langle u, \hat{M}_F^2 u \rangle \tag{2.7}$$

gives the eigenvalue of $\hat{M}_F^2$ which contains information of $\rho(X_1)$ (as shown in (2.4)). This motivates our architecture in Appendix A. While our choice of non-linearities is simpler than those in 2.6 and 2.7, our neural network architecture is still able to learn an approximation to the true moment inversion map, as demonstrated through experiments in Section 2.4.1.

2.3.2 Neural networks for cryo-EM

We make some alterations to our MRA architecture for cryo-EM reconstruction since we need to output a continuous representation of the volume to facilitate computing the moments

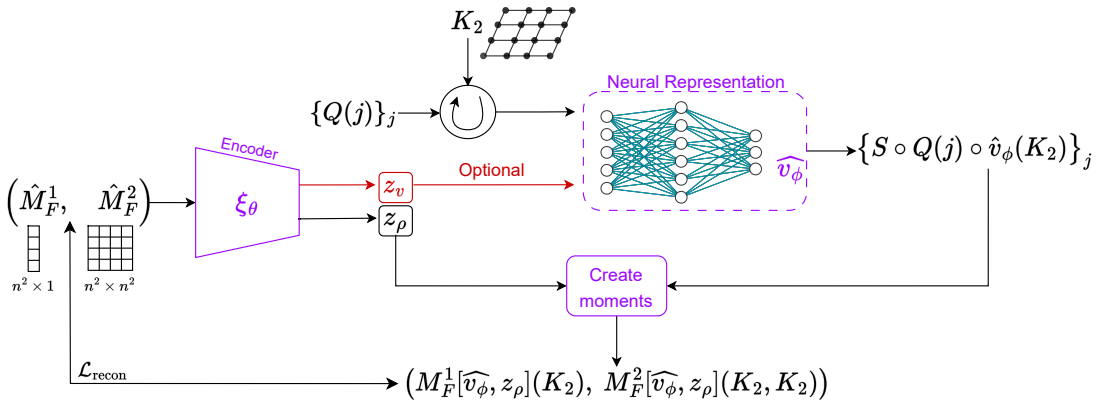

Figure 2.2: **Overview of our cryo-EM pipeline**: The encoder ξ_θ takes moments $(\hat{M}_F^1, \hat{M}_F^2)$ as input, and outputs $z_\rho \in \mathbb{R}^{|Q|}$, approximating a discretized probability density $\big(\rho(R)\big)_{R \in Q}$ for some fixed set of quadrature points $Q \subset \mathrm{SO}(3)$. Next, we create copies of the grid K_2 rotated corresponding to the elements of Q and input them to our neural representation $\widehat{v}_\phi$, which outputs corresponding slices of a running estimate of $\widehat{v}$. These slices $\Big\{S \circ Q(j) \circ \widehat{v}_\phi(K_2)\Big\}_j$ along with z_ρ are used to create $\Big(M_F^1[\widehat{v}_\rho, z_\rho](K_2), M_F^2[\widehat{v}_\rho, z_\rho](K_2, K_2)\Big)$ via equation (2.8), which we then compare with the inputs to the encoder, i.e., $(\hat{M}_F^1, \hat{M}_F^2)$ via the loss function $\mathcal{L}_{\text{recon}}$ in (2.17). Optionally, ξ_θ can also be used to output an extra z_v, a latent variable of $\widehat{v}$ that can be provided to $\widehat{v}_\phi$ as an input.

involving the reconstructed volume. Just as in the case of MRA, we have an encoder ξ_θ^ρ that outputs information regarding the density. More precisely, let $Q \subset \mathrm{SO}(3)$ be a set of quadrature points on SO(3) and $q = |Q|$. We want $\xi_\theta^\rho : (\hat{M}_F^1, \hat{M}_F^2) \to z_\rho$ ($\hat{M}_F^1, \hat{M}_F^2$) are estimators of (2.2)) where z_ρ should approximate $\big(\rho(R)\big)_{R\in Q}$ and ρ is a density on SO(3).

However, unlike the case of MRA, we now want to have a continuous representation of the Fourier volume. Let $\widehat{v}_\phi : \mathbb{R}^3 \to \mathbb{C}$ be an NN that represents a volume on the Fourier domain, and K_2 be n^2 equispaced points on $[-\pi, \pi]^2$. Suppose $\widehat{v}_\phi = \widehat{v}$ and $z_\rho = \rho(Q)$, one can evaluate $M_F^1[\widehat{v}, \rho](K_2, K_2), M_F^2[\widehat{v}, \rho](K_2, K_2)$ defined in (2.2) approximately via the quadrature rule

$$\begin{aligned} M_F^1[\widehat{v}_\phi, z_\rho](K_2) &\approx \sum_{j=1}^{q} z_\rho(j) S \circ Q(j) \circ \widehat{v}_\phi(K_2), \\ M_F^2[\widehat{v}_\phi, z_\rho](K_2, K_2) &\approx \sum_{j=1}^{q} z_\rho(j) \left(S \circ Q(j) \circ \widehat{v}_\phi(K_2)\right) \otimes \left(S \circ Q(j) \circ \widehat{v}_\phi(K_2)\right), \end{aligned} \tag{2.8}$$

where, by an abuse of notation, we think about $z_\rho = \rho(Q)$, i.e., the density ρ discretized on Q, as ρ itself and $Q(j)$ is an element in the set Q. For simplicity, in this chapter, we consider a quadrature rule with uniform quadrature weights, as seen in (2.8). The benefit of having a continuous $\widehat{v}_\phi$ is clear, since it allows us to obtain $\widehat{v}_\phi(Q(j)^T(k_x, k_y, 0))$ for any $(k_x, k_y) \in K_2$ easily.

Note that we also allow the flexibility to have an encoder ξ_θ^v just as in the case of MRA. In this case, $\xi_\theta^v : (\hat{M}_F^1, \hat{M}_F^2) \to z_v$ where z_v is some latent variable of the volume. In this case, we simply let $\widehat{v}_\phi : \mathbb{R}^{3+|z_v|} \to \mathbb{C}$ where the extra inputs of $\widehat{v}_\phi$ corresponds to the output of ξ_θ^v. The neural network pipeline we devise is shown in Figure 2.2, where $\xi_\theta = \left(\xi_\theta^\rho, \xi_\theta^v\right)$. As for the architecture of $\xi_\theta^v, \xi_\theta^\rho$, we adopt the type of architecture we use in Section 2.3.1, though one should be able to improve it according to the structure of the cryo-EM problem. The details of ξ_θ and $\widehat{v}_\phi$ are given in A.

2.4 Numerical examples

This section presents the results of numerical experiments with our method of moments algorithm with NN prior done using PyTorch [51].

2.4.1 Multireference alignment

We first present results using the method for MRA in Section 2.3.1. There are two phases when using the neural network detailed in Section 2.3.1:

- **Supervised training phase**: We are given a set of MRA datasets, each containing observations corresponding to a single signal and distribution of shifts, where the signal and shift distribution differs across the set of datasets. Our neural network is trained in a supervised manner to output the underlying signals and the distribution of shifts, with moment pairs of each dataset being the input.

- **Reconstruction phase**: A new and previously unseen dataset is provided, with observations corresponding to a single signal and distribution of shifts. Our neural network performs moment-matching with the empirical moments calculated from this dataset, using the loss function in 2.15, and outputs the underlying signal and distribution of shifts.

It is important to distinguish between the two phases since their settings are a little different, and the former can optionally be used to expedite convergence of the latter.

For evaluation purposes, we define the reconstruction error (also referred to as relative error) of an estimator $u \in \mathbb{R}^n$ of a signal v (or a distribution ρ) discretized at X_1, to be

$$\inf_{s\in\mathcal{I}} \frac{\left\|s \circ v\ (X_1)\ -\ u\right\|_F}{\left\|v\,(X_1)\right\|_F}, \quad \inf_{s\in\mathcal{I}} \frac{\left\|s \circ \rho\ (X_1)\ -\ u\right\|_F}{\left\|\rho\,(X_1)\right\|_F}. \tag{2.9}$$

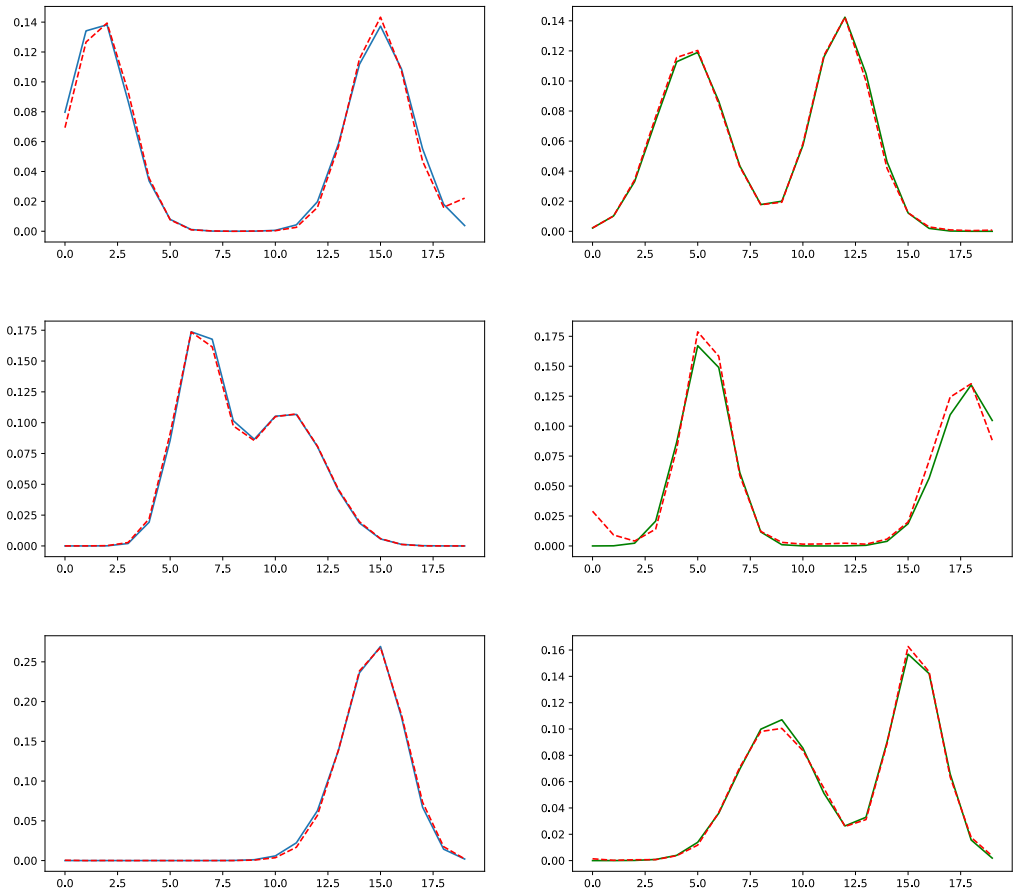

Figure 2.3: Predictions for the distribution ρ (Left) and volume v (Right), made by trained encoders ξ_θ^ρ and ξ_θ^v respectively, for ρ, v being mixture of 2 Gaussians. The solid lines are the ground truth ρ and v, while the dotted lines are the corresponding predictions by a neural network.

In addition, we define the relative errors for any moment estimators A_1, A_2 for the first and second moments, respectively, as

$$\frac{\left\| M_F^1[\widehat{v}, \rho](K_1) - A_1 \right\|_F}{\|A_1\|_F}, \quad \frac{\left\| M_F^2[\widehat{v}, \rho](K_1, K_1) - A_2 \right\|_F}{\|A_2\|_F}. \tag{2.10}$$

Supervised training Phase

In this section, we demonstrate that the moment inversion map can be learned by neural networks in a supervised way. To this end, we pre-select a distribution of signals $\mathcal{V}$ and a distribution of densities of shifts, $\mathcal{P}$. We draw N pairs of signal and shift density, $(v_1, \rho_1), (v_2, \rho_2), \ldots, (v_N, \rho_N)$ from $\mathcal{V} \times \mathcal{P}$. Using these pairs as our ground truths, we form their corresponding first and second moment pairs, i.e. $\Big(M_F^1[\widehat{v}_1, \rho_1](K_1), M_F^2[\widehat{v}_1, \rho_1](K_1, K_1)\Big)$, $\ldots, \Big(M_F^1[\widehat{v}_N, \rho_N](K_1), M_F^2[\widehat{v}_N, \rho_N](K_1, K_1)\Big)$. We then train our encoder ξ_θ in a supervised way to take inputs of the form $\Big(M_F^1[\widehat{v}_j, \rho_j](K_1), M_F^2[\widehat{v}_j, \rho_j](K_1, K_1)\Big)$ and output $\Big(\rho_j(X_1), \widehat{v}_j(K_1)\Big)$, for all $j \in \{1, 2, \ldots, N\}$.

In our experiments, we let both $\mathcal{V}$ and $\mathcal{P}$ be the family of mixtures of Gaussians on the interval $\mathcal{I}$, where we repeat our training procedure separately for a different number of Gaussians. We take 1.75×10^6 of input-output moment pairs to do the training using (2.9). We compute test error on 2.5×10^5 of samples.

We now discuss the hyperparameters for training. We train the encoders ξ_θ^ρ and ξ_θ^v separately; let us consider ξ_θ^ρ. We take the training set and feed the moments pairs $\Big(M_F^1[\widehat{v}_j, \rho_j](K_1)$, $M_F^2[\widehat{v}_j, \rho_j](K_1, K_1)\Big)$ to ξ_θ^ρ, which outputs corresponding z_ρ for each pair as a prediction for $\rho(X_1)$. We train ξ_θ^ρ over a total of 3×10^4 epochs with learning rates of $10^{-4}, 10^{-5}$ and 10^{-6} over 10^4 epochs successively. We then repeat the same process for ξ_θ^v.

Table 2.1 summarizes the average relative error on the training and test sets, using (2.9), while evaluating our trained encoders on mixtures of different numbers of Gaussians. The left and right columns of Figure 2.3 show some comparisons of the encoder output (z_ρ, z_v) with ground truth $(\rho(X_1), \widehat{v}(K_1))$ from the test set.

Reconstruction Phase

In the previous section, we discussed our process of training the encoder ξ_θ in a supervised way such that it learns the moment inversion map. A useful application of this trained

No. of Gaussians	z_ρ (Train error)	z_ρ (Test error)	z_v (Train error)	z_v (Test error)
1	0.042	0.048	0.048	0.052
2	0.121	0.141	0.156	0.170
3	0.177	0.195	0.180	0.206

Table 2.1: Average reconstruction errors (defined in (2.9)) of predictions z_ρ and z_v on training and test sets for mixtures of Gaussians.

encoder is when supplied with new, possibly noisy, moments $(\hat{M}_F^1, \hat{M}_F^2)$ from a single MRA dataset, we can use its outputs as a good initialization for further refinement. In this section, we demonstrate that this procedure leads to faster convergence.

We first talk about how we obtain the estimators $\hat{M}_F^1, \hat{M}_F^2$ from observations of the form

$$v_j = s_j \circ v(X_1) + \epsilon_j, \quad j = 1, \ldots, N \tag{2.11}$$

where $\epsilon_j \sim N(0, \sigma^2 I_n)$. Let $F \in \mathbb{C}^{n \times n}$ again be the Fourier matrix, we form unbiased moment estimators of the form

$$\hat{M}_F^1 = \frac{1}{N}\sum_{j=1}^{N} F v_j, \quad \hat{M}_F^2 \;=\; \frac{1}{N}\sum_{j=1}^{N} (Fv_j)(Fv_j)^* \;-\; \sigma^2 I_n \tag{2.12}$$

by subtracting a constant term on the diagonal of the empirical second moment. These are used as input to the trained encoder ξ_θ for prediction. Note that from 2.11 and 2.12, we get that as $N \to \infty$,

$$\begin{aligned} \hat{M}_F^1 &\xrightarrow{\text{a.s.}} \mathbb{E}(\hat{M}_F^1) \;=\; M_F^1[\widehat{v}, \rho](K_1, K_1) \\ \hat{M}_F^2 &\xrightarrow{\text{a.s.}} \mathbb{E}(\hat{M}_F^2) \;=\; M_F^2[\widehat{v}, \rho](K_1, K_1), \end{aligned} \tag{2.13}$$

due to the strong law of large numbers [20]. This means that even in very low SNR regime, with sufficiently high number of samples N, the empirical moments can be made as close to the analytical ones as desired. This underlines the noise resilience property of the method

of moments.

Notice that the solution to the MRA problem has a global translation ambiguity. Therefore, it is possible for the encoders $\xi_\theta^v, \xi_\theta^\rho$, to output an approximation to signal v and density ρ up to some arbitrary translations. While this is not an issue if the predicted signal is all we want, it becomes an issue if we want to refine the predictions further. More precisely, before deploying the encoder for refinement with empirical moments $\hat{M}_F^1, \hat{M}_F^2$ coming from a new dataset, we conduct an alignment across $s \in X_1$ to ensure that the outputs $z_\rho = \xi_\theta^\rho(\hat{M}_F^1, \hat{M}_F^2)$ and $z_v = \xi_\theta^v(\hat{M}_F^1, \hat{M}_F^2)$, upon forming

$$\begin{aligned} M_F^1[z_v, z_\rho](K_1) &= \sum_{j=1}^{n} z_\rho(j) \exp(-iK_1 s \odot z_v, \\ M_F^2[z_v, z_\rho](K_1, K_1) &= \sum_{j=1}^{n} z_\rho(j) \left(\exp(-iK_1 s(j)) \odot z_v\right) \left(\exp(-iK_1 s) \odot z_v\right)^*, \end{aligned} \quad (2.14)$$

match the inputs $(\hat{M}_F^1, \hat{M}_F^2)$ of the encoder. In other words, we loop over X_1 and select the shift s that minimizes the loss function

$$\mathcal{L}_{\text{recon}} = \left\| \hat{M}_F^1 - M_F^1[z_v, z_\rho](K_1) \right\|_F + \lambda \left\| \hat{M}_F^2 - M_F^2[z_v, z_\rho](K_1, K_1) \right\|_F, \quad (2.15)$$

then shift z_v by s to achieve the best alignment. Here by abuse of notation, we treat z_v, z_ρ as continuous objects and apply the functionals M_F^1, M_F^2 to them. Recall that $z_v = \xi_\theta^v(\hat{M}_F^1, \hat{M}_F^2)$ and $z_\rho = \xi_\theta^\rho(\hat{M}_F^1, \hat{M}_F^2)$. We further optimize the neural network parameters θ to refine z_v, z_ρ with the loss in (2.15). We refer to the process of minimization of this loss function as moment-matching or moment-fitting.

We now show the results of the deployment of our architecture ξ_θ when working with noisy moments from test datasets during the reconstruction procedure. We take 20 different moments pairs, i.e. $\left(\hat{M}_F^1, \hat{M}_F^2\right)$s corresponding to different datasets, and determine their corresponding (z_v, z_ρ) pairs by minimizing (2.15) over the parameters of ξ_θ. The relative errors

(defined in (2.9) and (2.10)) of the reconstructed $(\rho(X_1), \widehat{v}(K_1))$ and the moments are plotted in Figure 2.4. The errors are averaged over 20 different instances of (ρ, v) combinations from mixtures of 2 Gaussians, and the empirical moments are formed from 10^6 observations for each pair of (ρ, v) as in (2.11), with Gaussian noise $\sigma = 1.0$. Depending on whether the encoder underwent supervised training, we observe the trajectory of this "average" reconstruction error to be different. Figure 2.4 illustrates that the average reconstruction error indeed converges faster when the encoder is trained in a supervised phase.

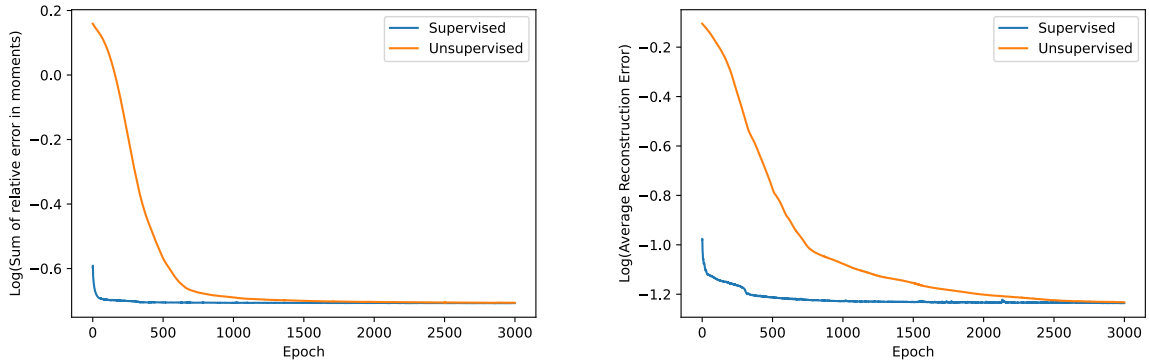

Figure 2.4: Plots of logarithms (with base 10) of Sum of relative errors (defined in (2.10)) for $\hat{M}_F^1$ and $\hat{M}_F^2$ across 3000 iterations (Top), and Reconstruction error (defined in (2.9)) across 3000 iterations (Bottom); averaged over 20 reconstructions of $(\rho(X_1), \widehat{v}(K_1))$ pairs drawn from the family of a mixture of 2 Gaussians. In both plots, the blue curve corresponds to the scenario where the encoder underwent supervised training, while the orange corresponds to the scenario where it did not.

2.4.2 Cryo-EM

We now present the results using our method for cryo-EM as illustrated in Section 2.3.2. Again for evaluation purposes, the relative error for an estimate $u \in \mathbb{R}^{n^3}$ of a signal v discretized at n^3 equispaced points X_3 on $\mathcal{I}^3$, is defined as

$$\inf_{R \in \mathrm{SO}(3)} \frac{\left\| R \circ v\ (X_3)\ -\ u \right\|_F}{\left\| v\,(X_3) \right\|_F}. \tag{2.16}$$

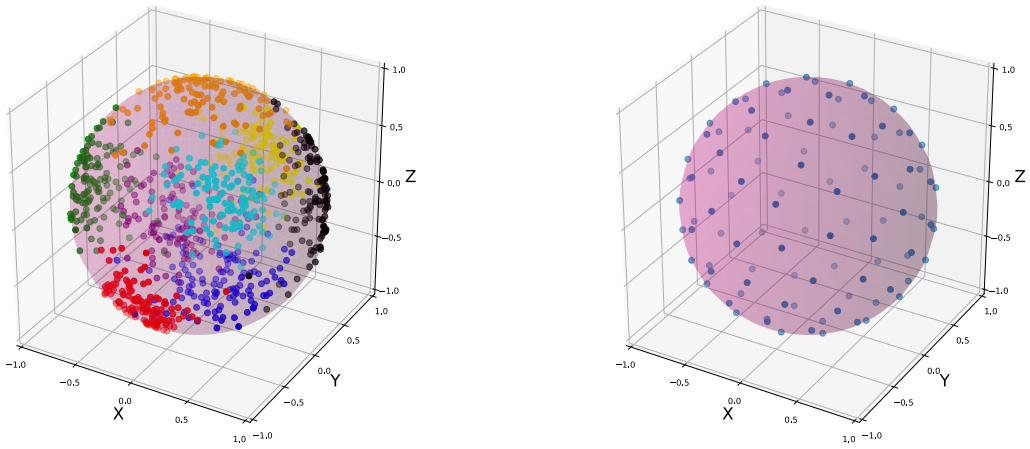

Figure 2.5: (Left) 1000 points sampled from a mixture of eight von Mises-Fisher random variables shown in different colors, and (Right) 100-point 13-design plotted on a 3D unit sphere.

The relative errors for moment estimators of the first and second moments are defined analogously to (2.10).

While we do not describe any supervised training phase like in the MRA case, our architecture keeps this option open. We believe that even for cryo-EM, it would be possible to train our encoder ξ_θ^ρ in a supervised way to learn the moment inversion map, i.e., to take inputs of the form $\left(M_F^1[\widehat{v},\rho](K_2), M_F^2[\widehat{v},\rho](K_2,K_2)\right)$ and predict $\big(\rho(R)\big)_{R\in Q}$ for training and reconstruction, where Q is the set of quadrature points on SO(3) defined in 2.3.2. It would also be possible to train ξ_θ^v such that it outputs a discretized approximation of the volume from the moments, or at least some vector containing important feature information about it.

The reconstruction is carried out by optimizing the NN parameters θ and ϕ of our encoder $z_\rho = \xi_\theta^\rho(\hat{M}_F^1, \hat{M}_F^2)$ and neural representation $\widehat{v}_\phi$, respectively, to minimize the loss function

$$\mathcal{L}_{\text{recon}} \;=\; \left\|\hat{M}_F^1 \;-\; M_F^1[\widehat{v}_\phi, z_\rho](K_2)\right\|_F \;+\; \lambda\left\|\hat{M}_F^2 \;-\; M_F^2[\widehat{v}_\phi, z_\rho](K_2,K_2)\right\|_F. \qquad (2.17)$$

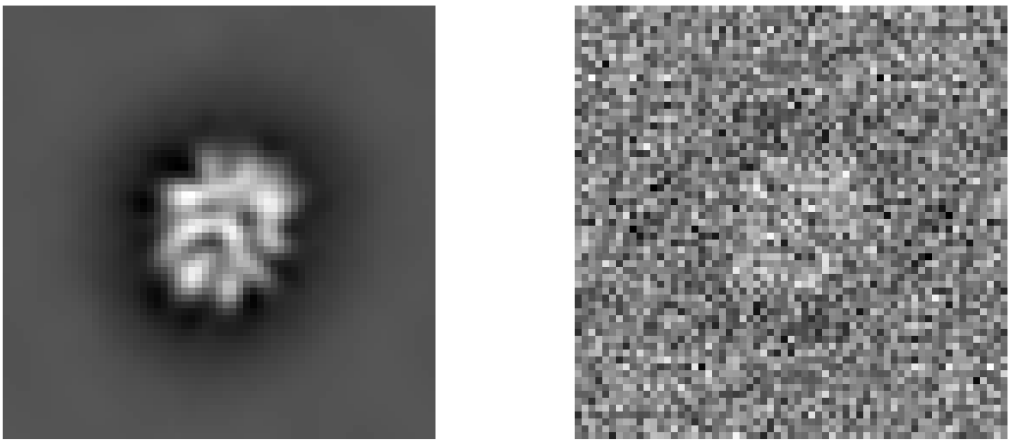

Figure 2.6: (Left) A clean projection, and (Right) its noisy counterpart with noise level $\sigma = 0.5$ as defined in (2.18), for EMD-0409

During reconstruction, one of the challenges we face is fixing a good set $Q \subset \mathrm{SO}(3)$ on which we shall use a quadrature rule with uniform weights to evaluate the functionals M_F^1, M_F^2, as described in (2.8). In our experiments, we do so in two steps. First, we choose a q_1-point spherical design on S^2, see, e.g., [75]. A q_1-point spherical t-design is a finite set of points with cardinality q_1 on S^2, such that their quadrature over S^2 with uniform unit weights is exact for any polynomial (spherical harmonics) with degree $\leq t$. Then, for each point of the design, treating the axis connecting that point to the center as a *viewing direction*, we consider in-plane rotations with q_2 equally spaced angles in $[0, 2\pi)$ radians. This gives us a set Q with $|Q| = q_1 q_2$ quadrature points on SO(3). In our experiments, we take $q_1 = 100$ and $q_2 = 12$ for a total of $|Q| = 1200$ quadrature points. To illustrate these quadrature points, we use a 100-point 13-design on S^2 as the set of viewing directions, as seen in the right side of Figure 2.5.

We now discuss our data generation process for cryo-EM and the moment estimators to

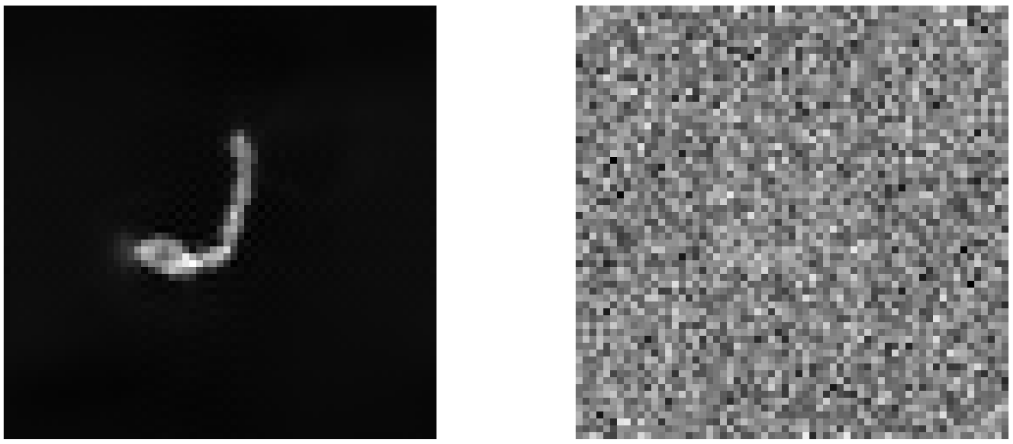

Figure 2.7: (Left) A clean projection, and (Right) its noisy counterpart with noise level $\sigma = 0.5$ as defined in (2.18), for EMD-25892

be used as input for the encoder. In practice, given real observations of the form

$$v_j = \mathcal{P} \circ R_j \circ v(X_2) + \epsilon_j, \quad j = 1, \ldots, N \tag{2.18}$$

where $\epsilon_j \sim N(0, \sigma^2 I_{n^2})$ and X_2 is n^2 equispaced points on $\mathcal{I}^2$, we could form unbiased moment estimators

$$\hat{M}_F^1 = \frac{1}{N} \sum_{j=1}^{N} F_2 v_j, \quad \hat{M}_F^2 \; = \; \frac{1}{N} \sum_{j=1}^{N} (F_2 v_j) \otimes (F_2 v_j) \; - \; \sigma^2 I_{n^2}, \tag{2.19}$$

letting $F_2 \in \mathbb{C}^{n^2 \times n^2}$ be the two-dimension Fourier transform matrix. Noiseless observations v_j are depicted alongside their noisy counterparts in Figures 2.6 and 2.7.

We next discuss our choices of ground truth volumes v and rotational distributions ρ. For our experiments, we use three volumes: EMD-0409 and EMD-25892 taken from the Electron Microscopy Data Bank (EMDB); and a mixture of four Gaussians not lying on the same plane in three dimensions. The dimensions of EMD-0409 are $128 \times 128 \times 128$ with voxel size

1.117 Å, while the dimensions of EMD-25892 are $320 \times 320 \times 320$ with voxel size 1.68 Å. Both volumes were downsampled to $63 \times 63 \times 63$ and scaled to have norm 1. The mixture of Gaussians has dimensions $25 \times 25 \times 25$, whose voxel size is taken to be 1 Å since it is a simulated volume. We represent the ground truth using $\widehat{v}_\phi$, and report the approximation error (as defined in (2.16)) between the original and this neural network representation, as 0.043 for EMD-0409, 0.076 for EMD-25892, and 0.004 for the mixture of Gaussians. These NN-approximated volumes are then used as the ground truths for the rest of the simulations, and we refer to them as the *neural* ground truths. The ground truth distribution of rotations ρ is chosen in the following way. The viewing directions are distributed as a mixture of 8 von Mises-Fisher distributions with different mean directions μ and concentration parameters κ, respectively, to ensure a sufficiently non-uniform distribution on S^2. 1000 points from this distribution are shown on the left side of Figure 2.5. The in-plane rotations are uniform on $[0, 2\pi)$ and independent of the viewing directions. We then create moment estimators from $N = 5 \times 10^6$ noisy observations with noise level $\sigma = 0.5$ using (2.19), where a neural slice approximates $F_2 v_j$.

We run our algorithm with learning rates 10^{-5} and 10^{-6} successively for $10,000$ epochs each, to minimize the loss function in (2.17). The reconstructed volumes are visualized in Figures 2.8, 2.9, and 2.10, alongside their corresponding neural ground truth volumes for EMD-0409, EMD-25892, and mixture of Gaussian volumes, respectively. Table 2.2 shows the relative errors of our moments from the reconstructed volumes, defined analogously to (2.10), at the end of our reconstruction.

Finally to evaluate the quality of reconstruction, we first align the reconstructed volumes with the ground truth. For that purpose, we run the algorithm for aligning three-dimensional density maps in [28] multiple times and pick the best alignment. We then calculate the Fourier Shell Correlation (FSC) between the ground truth volumes and their corresponding aligned reconstructions. We denote the resolution of the reconstructed volume as the point

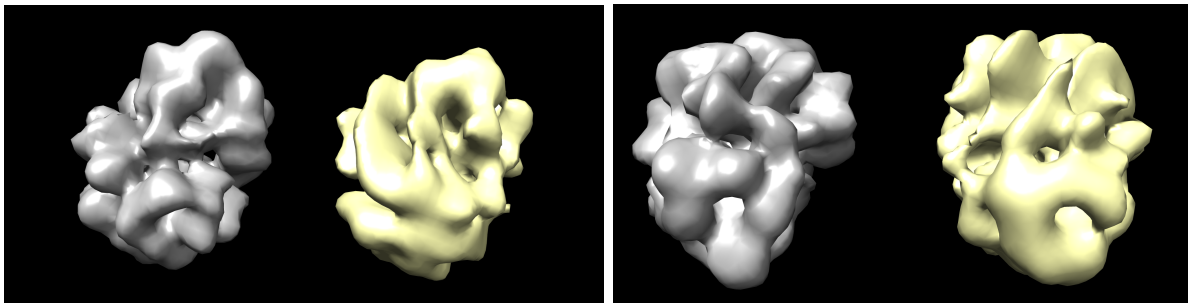

Figure 2.8: Ground truth volume (in gray) and reconstructed volume (in yellow) for the EMD-0409 volume, visualized using UCSF Chimera [53].

where the FSC curve goes below 0.5. The final resolutions between the ground truths and reconstructed volumes are provided in Table 2.3.

Volume	Relative error in $\hat{M}_F^1$	Relative error in $\hat{M}_F^2$
EMD-0409	0.003	0.013
EMD-25892	0.007	0.035
Mixture of Gaussians	0.007	0.016

Table 2.2: Final relative errors of moment estimates $\hat{M}_F^1$ and $\hat{M}_F^2$ after reconstruction phase.

Volume	Resolution (in Å)
EMD-0409	16.86
EMD-25892	21.52
Mixture of Gaussians	4.45

Table 2.3: Optimal resolutions between ground truth volumes and their reconstructions.

2.5 Conclusion and outlook

In this chapter, we addressed the reconstruction problem in cryo-EM as well as one of its simpler versions, namely, multirefence alignment, both of which fall under the class of orbit recovery problems. Although deep NN-based methods have been successfully used in maximum likelihood estimation for orbit recovery problems, they have not historically

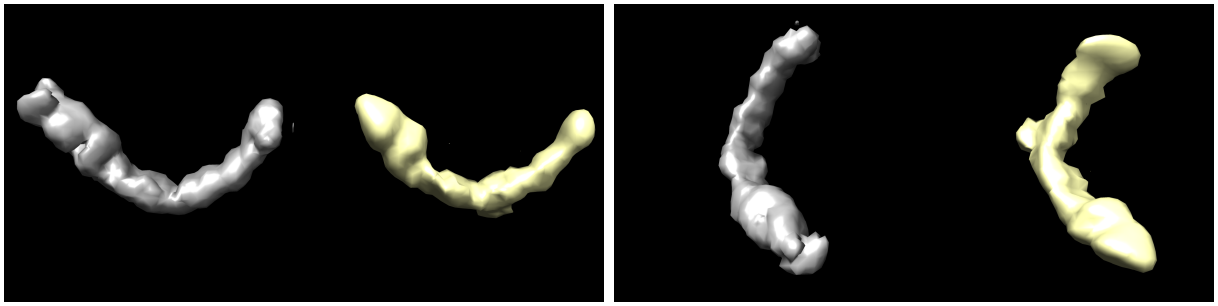

Figure 2.9: Ground truth volume (in gray) and reconstructed volume (in yellow) for the EMD-25892 volume, visualized using UCSF Chimera [53].

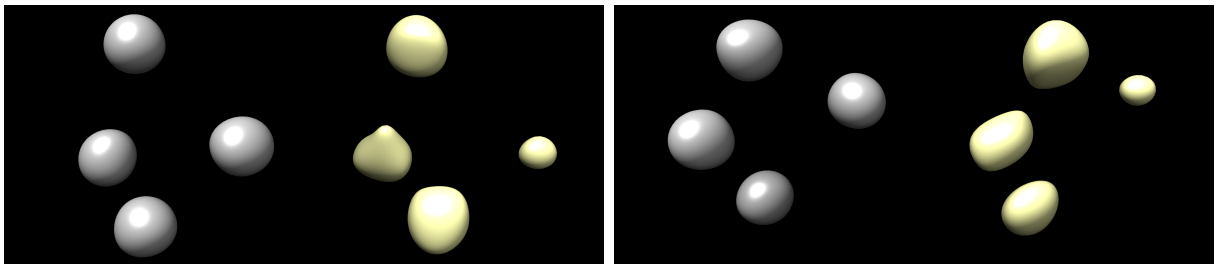

Figure 2.10: Two views of recovery of a mixture of Gaussians. Ground truth volume (in gray) and reconstructed volume (in yellow) for a mixture of 4 Gaussians in three dimensions, visualized using UCSF Chimera [53].

exploited the benefits offered by the MoM, like noise resilience, due to the central limit theorem when averaging data. In this chapter, we take a first step towards using neural networks for solving moment systems in orbit recovery problems. In the case of MRA, we demonstrate theoretically and numerically that a map can be learned to take moments as input and output the signal and density of translations, and develop novel neural network architectures for the same. This map can then be used as a deep neural network prior to accelerating convergence in unsupervised reconstruction from new incoming moments.

We also apply this approach to cryo-EM with encouraging results, but further work is needed to demonstrate the superiority of supervised learning and tackle more general cryo-EM models, like those dealing with small translations in addition to the rotations, and further image contamination due to aberrations (which would involve accounting for contrast transfer functions). Supervised learning would effectively enable low-dimension

reconstruction of volumes near-instantly and would serve as an inexpensive and time-efficient method of generating *ab-initio* models for iterative refinement algorithms. Other future work includes investigating the use of higher-order moments to improve reconstruction accuracy and parallelizing the model on multiple GPUs to enable reconstruction with larger images and improve speed and accuracy. Additionally, tackling more general cryo-EM models will bring us closer to operating on real-world datasets.

CHAPTER 3

MOM-NET: LEARNING CRYO-EM VOLUMES VIA METHOD OF MOMENTS

3.1 Introduction

Cryo-EM allows biologists to examine the structure of macromolecules in their natural state. Unlike the older method of X-ray crystallography, cryo-EM does not need crystallized samples, making it possible to study larger and more complex molecules with intricate structures and conformations. The scientific community has recognized its potential for revealing macromolecular structure and function: Nature Methods named cryo-EM the "Method of the Year" in 2015, and its development earned the 2017 Nobel Prize in Chemistry.

As mentioned in Chapter 1, the typical cryo-EM single particle reconstruction (SPR) workflow involves freezing a biological sample in a thin ice layer and imaging it with an electron microscope. The resulting images capture multiple copies of a macromolecule in various random and unknown orientations. Through several data processing steps, two-dimensional projections of the macromolecule's electrostatic potential are obtained in a series of images known as particle images. To prevent damage to the biological sample from the electron beam, imaging is performed at a low dosage, resulting in a poor signal-to-noise ratio (SNR). The primary objective of the cryo-EM SPR workflow is to reconstruct a three-dimensional volume representing the molecule's structure from these particle images.

In Chapter 2, we devised a novel neural network-based method for solving the SPR problem in cryo-EM, and recovering an *ab-initio* 3D volume from 2D projections. However, our simplified setting where the ground truth volume was rotated by a random element of SO(3) and projected, is rather limiting and non-representative of real data. In reality, each projection of the volume is slightly offset from the image center by an unknown amount, and the particle images are further blurred by a contrast transfer function (CTF), which is

specific to each image and depends on the microscope settings. There is also the issue of conformational heterogeneity, i.e. the projection images could correspond to different underlying volumes, with minor conformational differences between them. Inability to separate the conformational states properly results in low-resolution volume recovery. Consequently, cryo-EM reconstruction involves solving a tomography problem with structural variability, unknown viewing directions, in-plane shifts, and low SNR. The particle orientations and in-plane shifts together are referred to in literature as the pose variables.

In this chapter we introduce MoM-net, an extension of our cryo-EM framework from Chapter 2 which we have adapted to serve a two-fold purpose. Firstly, when provided with a set of cryo-EM datasets, each containing observations corresponding to a single volume, distribution of rotations and shifts, MoM-net can be trained in a supervised manner to predict very low-resolution structures along with distributions of rotations and levels of shift present in the datasets, with the corresponding first and second moments as inputs. This extends the supervised learning results of MRA from Chapter 2 to cryo-EM. Furthermore, the data is allowed to come from a setting where the images are not perfectly centered. The other utility of MoM-net is the reconstruction step, where our model can reconstruct a refined 3D volume in presence of shifts. Again, similar to the MRA case in Chapter 2, we also demonstrate the ability to accelerate the reconstruction process of MoM-net when it is trained in a supervised manner beforehand. Our method thus enjoys all the advantages offered by the method of moments, like its robustness in mitigating the impact of noise, and not requiring multiple data passes which helps in dealing with large datasets, then uses that for *ab-initio* SPR in presence of shifts. This *ab-initio* model serves as an initialization for iterative refinement algorithms in our general cryo-EM setting, taking us a step closer to handling real-world datasets.

3.2 Method of moments for cryo-EM in presence of shifts

In this section we describe the more general SPR setting, and provide formulae for the analytical moments of data for the same. From Chapter 1, recall the simplified cryo-EM setting (1.11) in the absence of shifts, where the observations were given by

$$v_j = \mathcal{P} \circ R_j \circ v, \quad j = 1, \ldots, N, \tag{3.1}$$

where $R_j \in \mathrm{SO}(3)$ are the unknown group elements. In practice during the image generation process, the projected volume may not be perfectly centered. Instead, they could be translated slightly from the center of the image in any random direction. The image formation model then becomes

$$v_j = t_j \circ \mathcal{P} \circ R_j \circ v, \quad j = 1, \ldots, N, \tag{3.2}$$

where $t_j \in \mathbb{R}^2$ are i.i.d 2D translations that are also independent of the rotations. By abuse of notation, we define the action of t_j on a 2D function $g \in L^1(\mathbb{R}^2)$ as

$$t_j \circ g \;=\; g(. - t_j). \tag{3.3}$$

Again, the Fourier domain formulation becomes

$$\widehat{v}_j(k_x, k_y) \;=\; \exp(ik^T t_j)\; S \circ R_j \circ \widehat{v}\;(k_x, k_y), \quad k = (k_x, k_y) \in [-\pi, \pi]^2, \tag{3.4}$$

where $\widehat{v}_j$ is the Fourier transform of v_j. We assume that t_j are i.i.d samples from bivariate gaussian distribution ψ with zero mean and diagonal covariance matrix $\eta^2 I_2$. Thus, η is the only additional parameter we need to estimate in this setting where $t_j \sim \psi = N(0, \eta^2 I_2)$.

The first and second analytical moments in this scenario are denoted by

$$\begin{aligned} M_F^1[\widehat{v},\rho,\eta](k_x,k_y) &= \mathbb{E}_{\rho,\psi}\left(\frac{1}{N}\sum_{j=1}^{N}\widehat{v}_j(k_x,k_y)\right) \\ M_F^2[\widehat{v},\rho,\eta](k_x,k_y,k_x',k_y') &= \mathbb{E}_{\rho,\psi}\left(\frac{1}{N}\sum_{j=1}^{N}\widehat{v}_j(k_x,k_y)\widehat{v}_j(k_x',k_y')^*\right). \end{aligned} \tag{3.5}$$

Notice that since $\psi = N(0,\eta^2 I_2)$, the first moment becomes

$$\begin{aligned} M_F^1[\widehat{v},\rho,\eta](k_x,k_y) &= \mathbb{E}_{t\sim\psi}\left[\mathbb{E}_{R\sim\rho}\left[\exp(ik^Tt)\ S\circ R\circ\widehat{v}\ (k_x,k_y)\mid t\right]\right] \\ &= \mathbb{E}_{t\sim\psi}\left[\exp(ik^Tt)\right]\mathbb{E}_{R\sim\rho}\left[S\circ R\circ\widehat{v}\ (k_x,k_y)\right] \\ &= M_F^1[\widehat{v},\rho](k_x,k_y)\ \frac{1}{2\pi\eta^2}\int_{\mathbb{R}^2}\exp(ik^Tt)\exp\left(-\frac{\|t\|^2}{2\eta^2}\right)\ dt \\ &= M_F^1[\widehat{v},\rho](k_x,k_y)\ \frac{1}{2\pi\eta^2}\int_{\mathbb{R}^2}\exp\left(-\frac{\|t\|^2-i2\eta^2k^Tt}{2\eta^2}\right)\ dt \\ &= M_F^1[\widehat{v},\rho](k_x,k_y)\exp\left(-\frac{\eta^2}{2}\|k\|^2\right)\frac{1}{2\pi\eta^2}\int_{\mathbb{R}^2}\exp\left(-\frac{\left\|t-i\eta^2k^Tt\right\|^2}{2\eta^2}\right)dt \\ &= M_F^1[\widehat{v},\rho](k_x,k_y)\exp\left(-\frac{\eta^2}{2}\|k\|^2\right), \end{aligned} \tag{3.6}$$

where the final equality follows from the Cauchy's residue theorem [37]. $M_F^1[\widehat{v},\rho]$ (given by (2.2)) is the associated first moment of the scenario where we have no translations (or equivalently, $\eta = 0$), and $k = (k_x, k_y)$. A similar calculation in case of second moment gives

$$M_F^2[\widehat{v},\rho,\eta](k_x,k_y,k_x',k_y') \ = \ M_F^2[\widehat{v},\rho](k_x,k_y,k_x',k_y')\ \exp\left(-\frac{\eta^2}{2}\left\|k-k'\right\|^2\right), \tag{3.7}$$

where $M_F^2[\widehat{v},\rho](k_x,k_y,k_x',k_y')$ (given by (2.2)) is the associated second moment in the no-translation case, and $k = (k_x,k_y)$ and $k' = (k_x',k_y')$. We aim to retrieve $\widehat{v}$ by matching

the moments $M_F^1[\hat{v}, \rho, \eta](K_2, K_2)$ and $M_F^2[\hat{v}, \rho, \eta](K_2, K_2)$ with some unbiased estimators $\hat{M}_F^1, \hat{M}_F^2$ when having noisy data.

3.3 Neural network prior for cryo-EM

This section presents neural network approaches for reconstructing the volume v and distribution ρ along with the shift standard deviation η, in the more general cryo-EM setting. The strategy is again to view both the volume, distribution and shift standard deviation as being mapped by a neural network from the estimated moments $\hat{M}_F^1$ and $\hat{M}_F^2$. As mentioned in Chapter 2, previous works have shown that the first 2 moments $\hat{M}_F^1, \hat{M}_F^2$ have enough information for recovering a low-resolution reconstruction [66] of the volume in the no-shift setting. In presence of shifts, (3.6) and (3.7) give us reason to believe that the first 2 moments will still possess enough information about the underlying volume. On that note, we extend the encoder-decoder structure of Chapter 2 to develop a new framework called MoM-net, that allows us to take the moments as input and give a continuous representation of a 3D volume.

The primary component of MoM-net is an encoder, i.e., a neural network ξ_θ, whose purpose is as follows. We are given a set of N datasets $\{\mathcal{D}_1, \ldots, \mathcal{D}_N\}$ corresponding to different combinations of volumes, distributions of rotations, and levels of shifts, i.e. $\{(v_1, \rho_1, \eta_1), \ldots, (v_N, \rho_N, \eta_N)\}$. Then for any $j \in \{1, \ldots, N\}$, the neural network ξ_θ takes the empirical moments $(\hat{M}_F^1, \hat{M}_F^2)_j$ corresponding to dataset $\mathcal{D}_j$ as inputs, and outputs vectors z_ρ, z_v, and $\hat{\eta}$ containing information about the distribution of rotations, the volume, and level of shifts respectively. In other words,

$$\xi_\theta : (\hat{M}_F^1, \hat{M}_F^2) \to (z_\rho, z_v, \hat{\eta}) \tag{3.8}$$

We explain the three outputs individually:

- Let $Q \subset \mathrm{SO}(3)$ be a pre-chosen set of quadrature points on SO(3) and $q = |Q|$. ρ is a density on SO(3), and $z_\rho \in \mathbb{R}^q$ should approximate $\big(\rho(R)\big)_{R\in Q}$.

- We again output a continuous representation of the Fourier volume, but this time we do it in 2 steps in order to aid the supervised learning process. The spherical harmonic representation is a convenient way of representing the Fourier volume for SPR problems (see [66, 9]) since it is a steerable basis, i.e. a function space closed under rotations. Mathematically, the band-limited Fourier volume $\widehat{v}$ can be expanded to degree L as

$$\widehat{v}(k,\theta,\phi) \;\approx\; \sum_{l=0}^{L} \sum_{m=-l}^{l} \sum_{s=1}^{S(l)} A_{l,m,s} F_{l,s}(k) Y_l^m(\theta,\phi), \tag{3.9}$$

  where k is the radial frequency, and Y_l^m are complex spherical harmonics defined by

$$Y_l^m(\theta,\phi) \;=\; \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}} P_l^m(\cos\theta) e^{im\phi}, \tag{3.10}$$

  with P_l^m being Legendre polynomials. $F_{l,s}$ are spherical Bessel functions [4], which are eigenfunctions of the Laplacian on a closed ball with Dirichlet boundary condition. Because our goal is low-resolution modeling, we can limit L and S_l in order to reduce computational burden. Hence, $A_{l,m,s}$ are the parameters we need to estimate in order to recover the volume. Enumerating $A_{l,m,s}$ as a vector, we get a vector of length, say $\tilde{L}$, and that is the vector that our encoder ξ_θ outputs an estimate of in the form of $z_v \in \mathbb{C}^{\tilde{L}}$. We refer to the continuous Fourier volume corresponding to z_v as $\widehat{v}_{\mathrm{sph}}$, which can then be refined as a next step.

- $\hat{\eta} \in [0, \eta_{\max}]$ is the estimate of the standard deviation of the shift distribution corresponding the the given data. $\eta_{\max}$ is a pre-chosen maximum possible shift level.

When $\widehat{v}_{\text{sph}}$ is obtained via z_v, MoM-net moves on to the second step: refinement. We use a second neural network, $\widehat{v}_{\text{Fnet}} : \mathbb{R}^3 \to \mathbb{C}$ initialized to the zero volume, to represent the refinement in Fourier domain. We thus get a continuous representation of the Fourier volume $\widehat{v}_\phi : \mathbb{R}^3 \to \mathbb{C}$, given by

$$\widehat{v}_\phi \;=\; \widehat{v}_{\text{sph}} \;+\; \widehat{v}_{\text{Fnet}}. \tag{3.11}$$

This continuous representation $\widehat{v}_\phi$ still has the same benefit as our previous framework, since it helps facilitate the computation of moments involving the reconstructed volume. Suppose $\widehat{v}_\phi = \widehat{v}$ and $z_\rho = \rho(Q)$, then if K_2 is n^2 equispaced points on $[-\pi, \pi]^2$, then $M_F^1[\widehat{v}, \rho, \eta](K_2, K_2), M_F^2[\widehat{v}, \rho, \eta](K_2, K_2)$ can be defined in (3.5) approximately via the quadrature rule

$$\begin{aligned}
M_F^1[\widehat{v}_\phi, z_\rho, \hat{\eta}](K_2) &\approx \left[\sum_{j=1}^{q} z_\rho(j) S \circ Q(j) \circ \widehat{v}_\phi(K_2)\right] \odot \left[\exp\left(-\frac{\hat{\eta}^2}{2}\|k\|^2\right)\right]_{k \in K_2}, \\
M_F^2[\widehat{v}_\phi, z_\rho, \hat{\eta}](K_2, K_2) &\approx \left[\sum_{j=1}^{q} z_\rho(j) \left(S \circ Q(j) \circ \widehat{v}_\phi(K_2)\right) \otimes \left(S \circ Q(j) \circ \widehat{v}_\phi(K_2)\right)\right] \\
&\qquad \odot \left[\exp\left(-\frac{\hat{\eta}^2}{2}\left\|k - k'\right\|^2\right)\right]_{\substack{k \in K_2 \\ k' \in K_2}},
\end{aligned} \tag{3.12}$$

where $\odot$ is elementwise multiplication. By an abuse of notation we think of $z_\rho = \rho(Q)$, which is the density ρ discretized on Q, as ρ itself, and $Q(j)$ is an element in the enumeration Q. In (3.12), we again consider a quadrature rule with uniform quadrature weights. Clearly the continuous $\widehat{v}_\phi$ allows us to obtain $\widehat{v}_\phi(Q(j)^T(k_x, k_y, 0))$ for any $(k_x, k_y) \in K_2$ easily.

The neural network pipeline of MoM-net is depicted in Figure 3.1. The architecture of the encoder ξ_θ is motivated by Section 2.3.1 with slight adjustments, while its exact details along with those of $\widehat{v}_{\text{Fnet}}$ are provided in A.

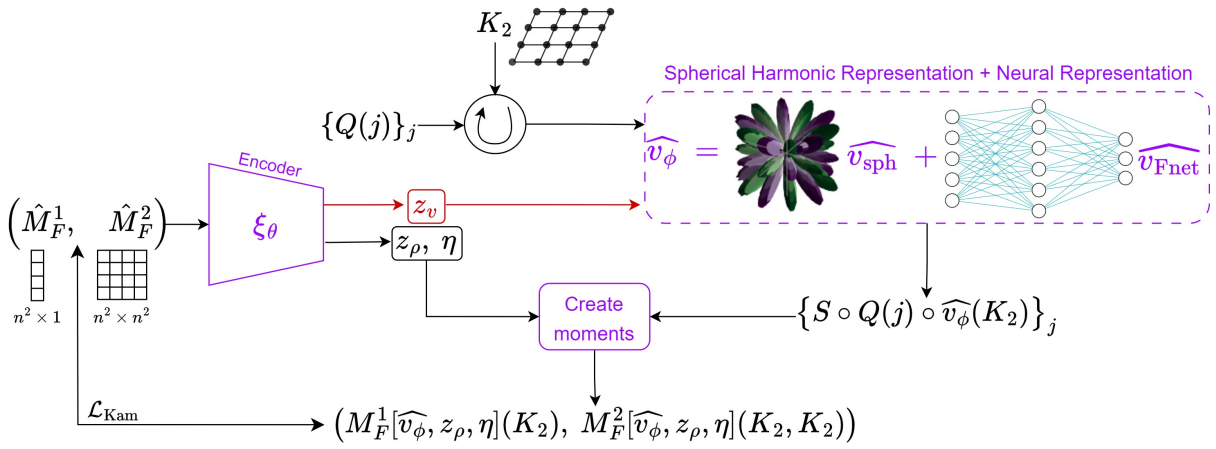

Figure 3.1: **Overview of MoM-net pipeline**: Our encoder ξ_θ takes moments $(\hat{M}_F^1, \hat{M}_F^2)$ as input, and outputs three things: $z_\rho \in \mathbb{R}^{|Q|}$ which approximates a discretized probability density $\big(\rho(R)\big)_{R \in Q}$ for some fixed set of quadrature points $Q \subset \mathrm{SO}(3)$; $z_v \in \mathbb{C}^{\tilde{L}}$ which approximates the spherical harmonics coefficients of the volume; and $\hat{\eta}$ which approximates the standard deviation of shifts in the images. Then we create copies of the grid K_2 rotated according to the elements of Q and input them to our continuous Fourier representation $\widehat{v}_\phi$, which is the sum of $\widehat{v}_{\text{sph}}$ (volume corresponding to z_v) and $\widehat{v}_{\text{Fnet}}$ (NN for refinement). $\widehat{v}_\phi$ outputs corresponding slices of a running estimate of $\widehat{v}$. These slices $\Big\{S \circ Q(j) \circ \widehat{v}_\phi(K_2)\Big\}_j$ along with z_ρ and $\hat{\eta}$ are used to create $\Big(M_F^1[\widehat{v}_\phi, z_\rho, \hat{\eta}](K_2), M_F^2[\widehat{v}_\phi, z_\rho, \hat{\eta}](K_2, K_2)\Big)$ via equation (3.12), which we then compare with the moments used as input for the encoder, i.e., $(\hat{M}_F^1, \hat{M}_F^2)$ via the loss function $\mathcal{L}_{\text{Kam}}$ in (2.17).

3.4 Numerical examples

3.4.1 Supervised training phase

For evaluation purposes, we define the relative error for an estimate $u \in \mathbb{R}^{n^3}$ of a volume v discretized at n^3 equispaced points X_3 on $\mathcal{I}^3$, and for moment estimators of the first and second moments as in Section 2.4.2. The aim of the supervised training phase is to use the encoder ξ_θ to predict (v, ρ, η) when provided with a previously unseen dataset of cryo-EM images. In other words, we demonstrate that the moment inversion map can be learned by neural networks in a supervised way for cryo-EM as well.

To this end, we pre-select a distribution of volumes $\mathcal{V}$, a distribution of densities of rotations $\mathcal{P}$, as well as a distribution of standard deviation of shifts $\mathcal{N}$. We draw N triplets of volumes, rotational densities, and standard deviations of shifts, say $(v_1, \rho_1, \eta_1), (v_2, \rho_2, \eta_2), \ldots,$ (v_N, ρ_N, η_N) from $\mathcal{V}\times\mathcal{P}\times\mathcal{N}$. Using these pairs as ground truths, we form their corresponding first and second moment pairs, i.e. $\left\{\left(M_F^1[\widehat{v}_j, \rho_j, \eta_j](K_1), M_F^2[\widehat{v}_j, \rho_j, \eta_j](K_1, K_1)\right)\right\}_{j=1}^{N}$. We then train our encoder ξ_θ in a supervised way to take inputs of the form $\Big(M_F^1[\widehat{v}_j, \rho_j, \eta_j](K_1),$ $M_F^2[\widehat{v}_j, \rho_j, \eta_j](K_1, K_1)\Big)$ and output $\left(z_v, z_\rho, \hat{\eta}\right)_j$, for all $j \in \{1, 2, \ldots, N\}$. Here, $z_\rho \in \mathbb{R}^{|Q|}$ approximates a discretized probability density $\Big(\rho_j(R)\Big)_{R\in Q}$ for some pre-chosen set of quadrature points $Q \subset \mathrm{SO}(3)$, $\hat{\eta}$ approximates η_j, while $z_v \in \mathbb{C}^{\tilde{L}}$ approximates the spherical harmonics corresponding to the degree L expansion of $\widehat{v}_j$.

In 2.4.1, we used the loss function (2.9) for supervised learning in case of MRA. This loss calculates the minimum relative error in the ground truth and predicted signal over each shift. Unfortunately, it would be a huge challenge to use that loss for cryo-EM, since that would require taking an infimum over the whole of SO(3). That would be computationally very intensive, even for small volumes. That is why in this case, we use a different loss which is a small variant of the Kam's volume metric introduced in [77]. We call it the Kam loss, and define it as

$$\mathcal{L}_{\text{Kam}} = \frac{\left\| M_F^1[\widehat{v},\rho,\eta](K_2) - M_F^1[z_v,z_\rho,\hat{\eta}](K_2) \right\|_F}{\left\| M_F^1[\widehat{v},\rho,\eta](K_2) \right\|} + \frac{\left\| M_F^2[\widehat{v},\rho,\eta](K_2,K_2) - M_F^2[z_v,z_\rho,\hat{\eta}](K_2,K_2) \right\|_F}{\left\| M_F^2[\widehat{v},\rho,\eta](K_2,K_2) \right\|}. \tag{3.13}$$

In other words, we define the Kam loss as the sum of relative errors between input and output moments. Thus when a moment pair is given to the encoder as input, we create another moment pair $\left(M_F^1[z_v,z_\rho,\hat{\eta}](K_2), M_F^2[z_v,z_\rho,\hat{\eta}](K_2,K_2)\right)$ using the outputs $(z_v,z_\rho,\hat{\eta})$ from (3.12), and compare them using (3.13).

In our experiments, we take $\mathcal{P}$ to be the family of mixtures of five von Mises-Fisher random variables with fixed mean vectors. $\mathcal{P}$ is taken to be a family of simulated protein-like structures having 20 atoms each, with isotropic Gaussian blobs on the atomic coordinates. Cryo-EM images of size 15×15 are then produced, and moments generated. Two cryo-EM images corresponding to two different volumes used for training, are provided in Figure 3.2. The distribution of standard deviation of shifts, i.e. $\mathcal{N}$, is taken to be uniformly distributed on $[0, 1.33]$. To save space, we generate different batches of input-output pairs for training, at every iteration of our optimization. We compute test error on 1.3×10^4 samples using (3.13).

We now discuss the hyperparameters for training. We train the encoder ξ_θ by feeding batches of moment pairs $\left(M_F^1[\widehat{v},\rho,\eta](K_2),\ M_F^2[\widehat{v},\rho,\eta](K_2,K_2)\right)$ to ξ_θ, which outputs corresponding $(z_v,z_\rho,\hat{\eta})$ for each pair, that are predictions for spherical harmonic coefficients of $\widehat{v}$, $\rho(Q)$ and η respectively. We train ξ_θ with batch sizes of 128 over a total of 2×10^4 epochs with learning rates of 10^{-5} and 10^{-6} over 10^4 epochs successively, by minimizing the Kam Loss (3.13). Our Kam loss on the test set comes out to be 0.056.

As a further test, for visualization purposes, we feed input moment pairs corresponding

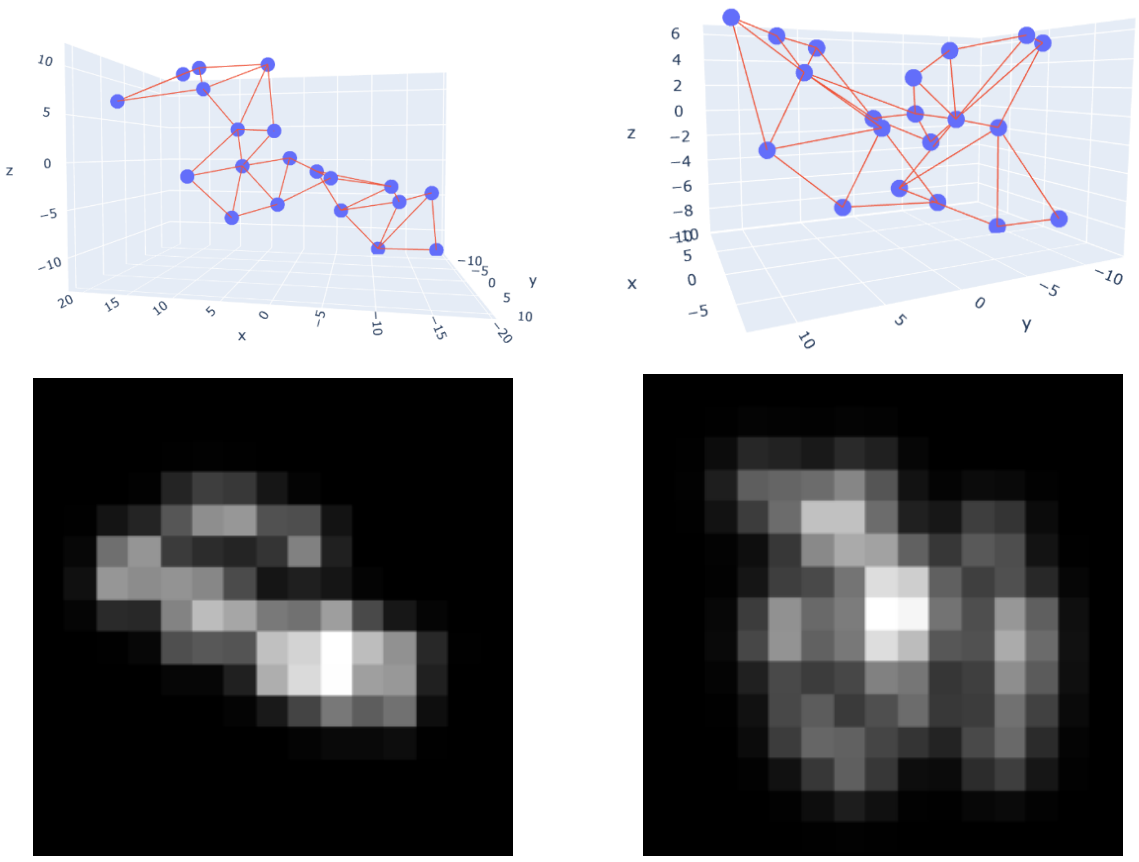

Figure 3.2: (Top row) Two different protein-like volumes used to train ξ_θ, and (Bottom row) two sample cryo-EM images corresponding to each of these volumes.

to the volumes EMD-0409 and EMD-25892 (used in Chapter 2) separately to the trained encoder. Note that neither of these structures fall in the family of volumes that ξ_θ was trained on. Their respective final Kam losses, along with their FSC values are provided in Table 3.1. While the predictions (displayed in Figures 3.3 and 3.4) are admittedly too low-resolution to be useful *ab initio* models, they do seem promising as good initializations that can be further refined to obtain higher resolution reconstructions in the next phase. Improving the quality of these predictions is a topic for further research.

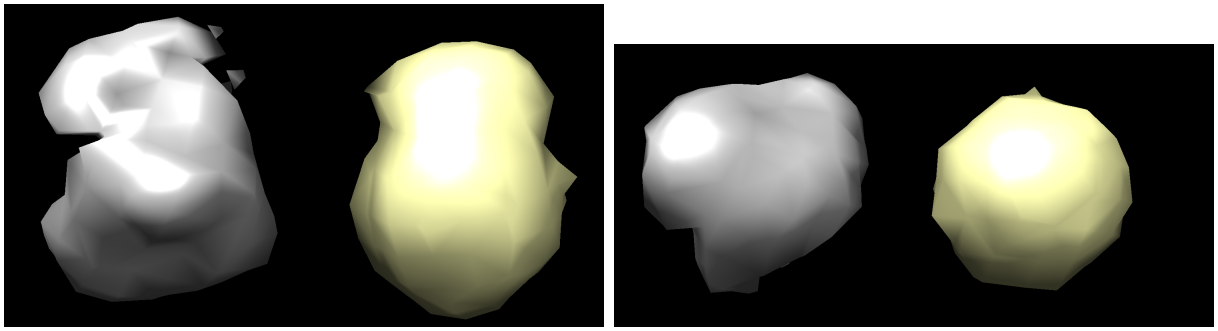

Figure 3.3: Two views of predicted volume by MoM-net. Reconstructed volume (in gray) and ground truth volume (in yellow) for EMD-0409, visualized using UCSF Chimera [53].

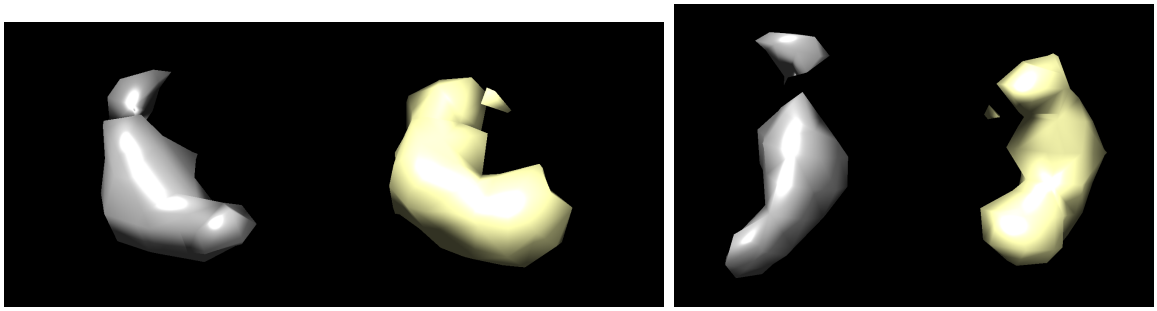

Figure 3.4: Predicted volume by MoM-net (in gray) and ground truth volume (in yellow) for EMD-25892, visualized using UCSF Chimera [53].

Volume	Final Kam loss	Resolution (in Å)
EMD-0409	0.094	74.27
EMD-25892	0.068	88.44

Table 3.1: Final Kam loss values of moment estimates $\hat{M}_F^1$ and $\hat{M}_F^2$, as well as optimal resolutions for ground truth volumes and predictions after supervised training phase.

3.4.2 Reconstruction phase

The reconstruction is carried out by optimizing the neural network parameters θ and ϕ of our encoder $(z_v, z_\rho, \eta) = \xi_\theta(\hat{M}_F^1, \hat{M}_F^2)$ and continuous representation $\widehat{v}_\phi$, respectively, to minimize the Kam loss. Here, $\hat{M}_F^1, \hat{M}_F^2$ are the first two moments from a previously unseen cryo-EM dataset corresponding to a single (v, ρ, η) triplet.

The setting of our reconstruction process is same as that of Section 2.4.2. We select our set of quadrature points $Q \subset \mathrm{SO}(3)$ by the same two step process: we pick a 100-point

13-design on S^2 as the set of viewing directions, and select in-plane rotations with 12 equally spaced angles in $[0, 2\pi)$ radians.

For our data generation process for cryo-EM: given real observations of the form

$$v_j = t_j \circ \mathcal{P} \circ R_j \circ v(X_2) + \epsilon_j, \quad j = 1, \ldots, N \tag{3.14}$$

where $\epsilon_j \sim N(0, \sigma^2 I_{n^2})$ and X_2 is n^2 equispaced points on $\mathcal{I}^2$, we form unbiased moment estimators

$$\hat{M}_F^1 = \frac{1}{N}\sum_{j=1}^{N} F_2 v_j, \quad \hat{M}_F^2 \;=\; \frac{1}{N}\sum_{j=1}^{N}(F_2 v_j) \otimes (F_2 v_j) \;-\; \sigma^2 I_{n^2}, \tag{3.15}$$

letting $F_2 \in \mathbb{C}^{n^2 \times n^2}$ be the two-dimension Fourier transform matrix. Our choice of ground truth volumes v is EMD-0409, downsampled to $45 \times 45 \times 45$ and scaled to have norm 1. The ground truth distribution of rotations ρ is also chosen the same way as in Section 2.4.2. We then create moment estimators from $N = 5 \times 10^6$ noisy observations with noise level $\sigma = 0.5$ using (3.15), where a neural slice approximates $F_2 v_j$.

We run both MoM-net as well as our previous framework of Chapter 2, with learning rates 10^{-5}, 10^{-6} and 10^{-7} successively for $2,000$ epochs each, to minimize the Kam loss (3.13). The entire optimization process is repeated separately for different datasets for the same volume and distribution of rotations, but with the standard deviation of shifts being $\eta = 0.0$ (depicted in Figure 3.5), $\eta = 2.0$, and $\eta = 4.0$ (both depicted in Figure 3.6) respectively. Table 3.2 shows the final Kam loss of our reconstructed volumes at the end of our reconstruction.

Clearly, the aforementioned figures along with Table 3.2 demonstrate the superiority of MoM-net over our previous framework, in presence of shifts in the data. We can see that MoM-net produces much better reconstructions since it can estimate the shift level as well, while our previous framework assumes the absence of shifts, hence producing much more

η	MoM-net	Framework in Chapter 2
0.0	0.012	0.013
2.0	0.008	0.014
4.0	0.007	0.017

Table 3.2: Comparison of final Kam loss for MoM-net and our framework from Chapter 2, after reconstruction phase, for three values of shift standard deviation η.

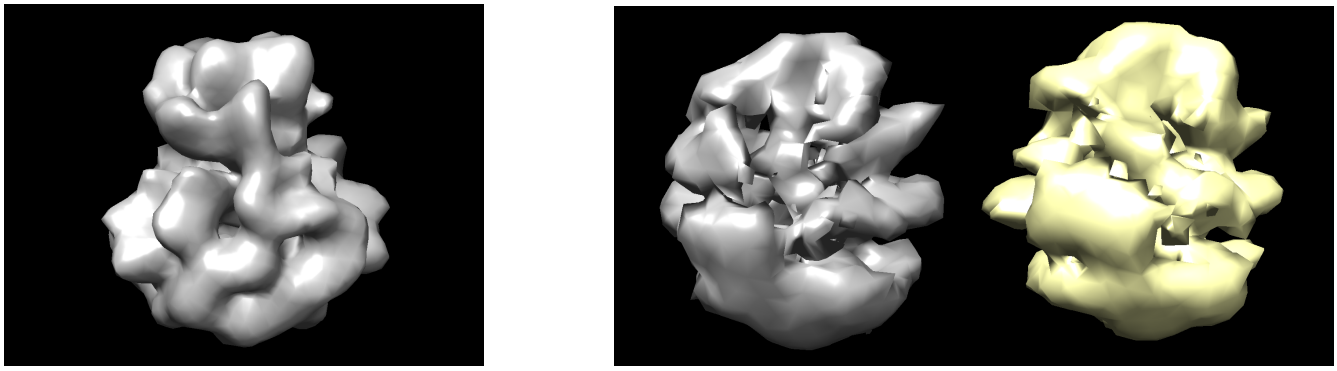

Figure 3.5: (Left) Ground truth volume EMD-0409. (Right) Volumes reconstructed using MoM-net (in gray) and using the old framework of Chapter 2 (in yellow) for EMD-0409, corresponding to shifts with $\eta = 0.0$. Images visualized using UCSF Chimera [53].

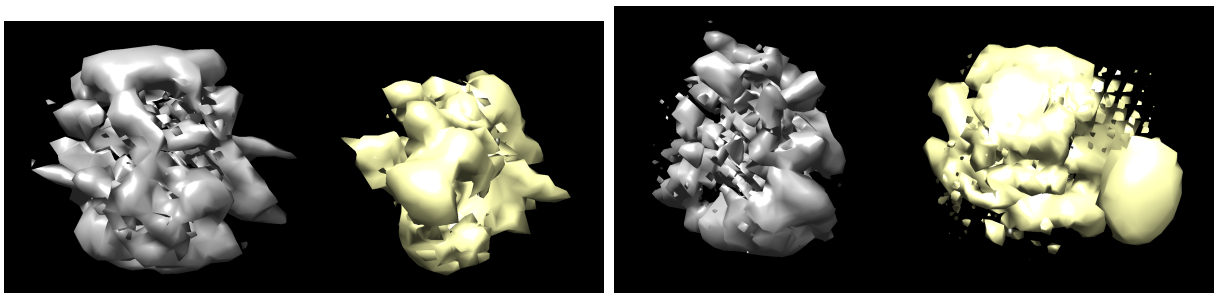

Figure 3.6: Volumes reconstructed using MoM-net (in gray) and using the old framework of Chapter 2 (in yellow) for EMD-0409, corresponding to shifts with $\eta = 2.0$ (Left) and $\eta = 4.0$ (Right) respectively. Images visualized using UCSF Chimera [53].

blurry and low-resolution reconstructions. While the reconstruction quality of MoM-net also suffers as η increases, but that is due to the loss of information regarding the volume that is contained in the first two moments.

Finally, we also show that training ξ_θ in a supervised manner results in slightly faster convergence than in the untrained scenario. We again choose the same volume and distri-

bution of rotations, with $\eta = 0.0$ and run MoM-net in two scenarios: trained and untrained. The logarithm of the corresponding Kam losses are plotted with respect to iterations in Figure 3.7. While not very impressive, the trained scenario does save some iterations in the reconstruction process.

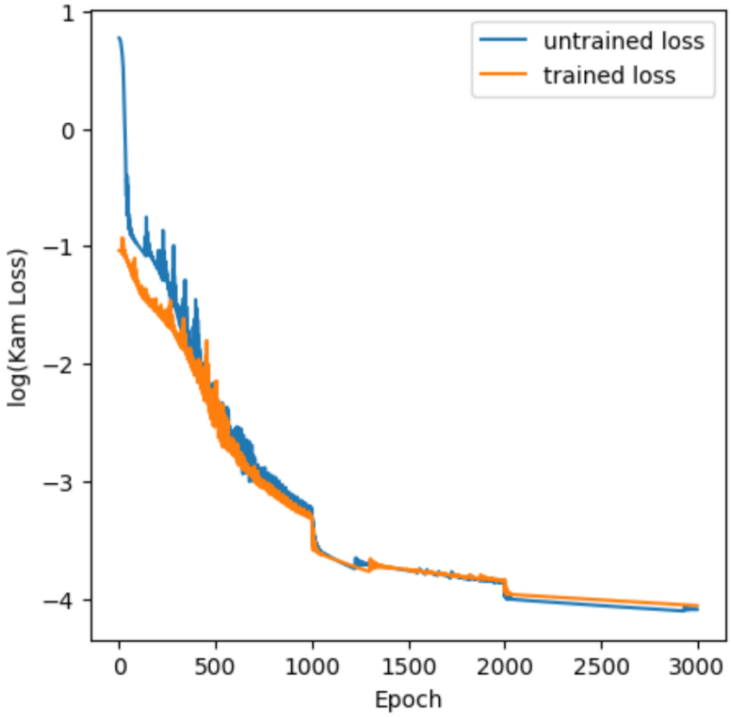

Figure 3.7: Plot of logarithm of Kam loss during reconstruction in the trained vs untrained scenarios, for EMD-0409 across 3000 iterations.

3.5 Summary

In this chapter, we have taken promising steps along two of the future lines of work underlined in Section 2.5, namely, supervised learning of volumes for cryo-EM, and volume recovery in a more general cryo-EM setting. Our neural network framework MoM-net takes the first step in the integration of neural networks for solving moment systems in cryo-EM, where we experimentally demonstrate that a map can be learned to take moments from a set of datasets as inputs and produce the volumes, distribution of rotations, as well as shift

variances as outputs. This gives us a near-instantaneous and cost-effective *ab-initio* estimate of the underlying volume (albeit very low-dimensional) simply from the moments of a new test dataset. We also show a potential usage of this map as a deep neural network prior to expedite convergence in unsupervised reconstruction provided new observed moments.

It would be worth investigating some future directions stemming from this work. Parallelizing our model across multiple GPUs could lead to improved speed and accuracy, and enable the handling of larger images. Third (and higher) order moments could be also used to improve reconstruction. Improving the quality of the predicted volume via supervised learning would also be extremely crucial, as this would lead to much faster convergence during the reconstruction process. A possible method for the same might be the usage of compressed moments during the training process, so that moments corresponding to larger sized images can be used.

While we have tackled the setting allowing for small shifts to be present in our data, real-world cryo-EM datasets also suffer from contamination due to aberrations, which require estimation of the CTF as well. Also present is conformational heterogeneity in the underlying volume. Advanced algorithms such as those implemented in software like RELION [44, 80] and cryoSPARC [56], are employed to distinguish between the different states and accurately reconstruct each one separately. This approach enables the detailed study of dynamic molecular machines and complexes, providing insights into their functional mechanisms and conformational changes. While MoM-net deals with homogeneous reconstruction for now, the ability to recover at least a finite number of conformations would be highly desirable. By capturing the structural heterogeneity inherent in biological samples, this improvement would significantly enhance the applicability of MoM-net to real-world datasets.

CHAPTER 4

ACCELERATING VARIANCE-REDUCED ERM AND EM ALGORITHMS IN ORBIT RECOVERY SETTING USING SECOND-ORDER INFORMATION

4.1 Introduction

In the last few chapters, we proposed methods to solve orbit recovery problems via MoM. In this chapter, we look at another formulation of orbit recovery, i.e. as latent variable models, which are an important class of models due to their wide applicability across machine learning and statistics [35, 13]. Expectation Maximization (EM) [17] is a useful tool that is widely used for maximum likelihood estimation for the parameters in latent variable models. It is an iterative algorithm with two steps: an E-step which calculates the expectation of sufficient statistics under the latent variable posteriors given the current parameters, and an M-step which updates the parameters given the expectations. Another technique that is often used is Empirical risk minimization (ERM), that directly minimizes the marginal likelihood function, usually by deploying some variant of gradient descent. Both EM and ERM can be accelerated using second-order information, with algorithms like Newton and quasi-Newton methods [54, 38].

With the advent of large-scale datasets and complex models, these classical techniques of ERM and EM often face challenges in terms of computational efficiency and scalability. Stochastic optimization algorithms, such as stochastic gradient descent (SGD) [61] and stochastic EM (sEM) [15], have gained popularity due to their ability to handle large datasets and efficiently optimize complex objective functions by sampling the dataset at every iteration. However, these methods may suffer from slow convergence rates and high variance in parameter updates, particularly in non-convex and ill-conditioned cases.

To address these limitations, stochastic variance-reduced gradient (SVRG) [31] methods

have been proposed as an effective alternative to traditional stochastic optimization algorithms in case of ERM. SVRG leverages additional information from past iterates to reduce the variance of stochastic gradients, thereby improving convergence rates and stability. Along a similar vein, stochastic variance-reduced EM (sEM-vr) [16] method was introduced as an alternative to traditional EM.

In this chapter, we analyse the usage of SVRG and sEM-vr along with their corresponding second-order accelerations, to solve two orbit recovery problems: windowed multireference alignment (MRA), and single particle cryo-EM modeling. SPA is a Nobel Prize-winning area [19], which is now considered a widely popular method for determining the atomic-resolution 3D structure of biological macromolecules, while windowed MRA is considered a simpler variant of the same [81]. We also propose a quasi-Newton method of accelerating sEM-vr and test it on simulated windowed MRA data.

4.2 The windowed multireference alignment model

Recall the general MRA model, (1.3), where $\mathcal{A}$ is the identity. Here, the unknown signal v is defined on a unit, symmetric segment $\mathcal{I} = [-\frac{1}{2}, \frac{1}{2}]$. In other words, the signal is $v : \mathcal{I} \to \mathbb{R}$, which is further assumed to be a periodic, band-limited function. Let G be the group of circular translations on $\mathcal{I}$, i.e. the group whose elements s_j shift v via

$$s_j \circ v := v(\cdot - s_j), \tag{4.1}$$

where the difference is interpreted as modulo the segment, namely $\cdot - s_j$ is always in $\mathcal{I}$. The MRA problem can also be reformulated in the Fourier domain. Assuming absence of noise, let $\widehat{v}_j$ be the Fourier transform of v_j. Then in the Fourier domain, the shift s_j becomes a

phase, i.e.

$$\widehat{v}_j(k) = \exp(iks_j)\widehat{v}(k),\ k \in [-\pi, \pi]. \tag{4.2}$$

The signal v_j is generally provided on n discretized points in $\mathcal{I}$, where n is chosen to satisfy its Nyquist frequency. Thus the frequency k has a natural bandlimit $|k| \leq \pi$. In Fourier domain, our observations become

$$\widehat{v}_j(K_1) = \exp(iK_1 s_j) \odot \widehat{v}(K_1), \tag{4.3}$$

where we define K_1 to be the set of n equispaced points in $[-\pi, \pi]$.

In case of windowed MRA, the operator $\mathcal{A}$ is the windowing operator, which curtails every general MRA observation to the first few positions. In real domain, let R_l be the transformation that shifts vectors in $\mathbb{R}^n$ by l positions, while $W_{\tilde{n}}$ windows it, i.e. discards all but the first $\tilde{n}$ positions. In matrix notation,

$$R_l = \begin{bmatrix} 0 & I_l \\ I_{n-l} & 0 \end{bmatrix}, \quad W_{\tilde{n}} = \begin{bmatrix} I_{\tilde{n}} & 0 \end{bmatrix}.$$

For the complex domain, define $F_n \in \mathbb{C}^{n \times n}$ as the matrix representation of a normalized Fourier transform where $F_n^* F_n = I_n$. Since the window length $\tilde{n}$ is fixed for a given problem, for ease of notation we define the complex windowing operator $\tilde{R}_l \in \mathbb{C}^{\tilde{n} \times n}$ as

$$\tilde{R}_l = F_{\tilde{n}} W_{\tilde{n}} R_l F_n^*.$$

For our observations in the windowed MRA model, let K_1^n and $K_1^{\tilde{n}}$ be the set of n and

$\tilde{n}$ equispaced points between $[-\pi, \pi]$ respectively. Then, we have

$$\widehat{v}_j(K_1^{\tilde{n}}) = \tilde{R}_l \widehat{v}_j(K_1^n). \tag{4.4}$$

4.3 Empirical Risk Minimization (ERM)

In this section we briefly explain methods of empirical risk minimization, which focuses on minimizing the marginal likelihood function directly. Let $f_1, f_2, \ldots, f_n$ be vector valued functions from $\mathbb{R}^d$ to $\mathbb{R}$. Then, the goal is to find

$$\min_{w \in \mathbb{R}^d} F(w), \qquad \text{where } F(w) = \frac{1}{n} \sum_{i=1}^{n} f_i(w)$$

For orbit recovery problems, the functions $f_1, f_2, \ldots, f_n$ are typically the likelihood functions corresponding to a dataset of n observations $v_1, v_2, \ldots, v_n$, i.e. for any $1 \leq i \leq n$, $f_i(w)$ is the model likelihood of parameter $w \in \mathbb{R}^d$ when v_i is observed.

4.3.1 Gradient descent

Gradient descent (GD) [61] is one of the foundational methods for unconstrained optimization, known for its simplicity. It follows the following update rule for $t \in \mathbb{N}$,

$$w_t \;=\; w_{t-1} - \eta_t \nabla F(w_{t-1}) \;=\; w_{t-1} - \frac{\eta_t}{n} \sum_{i=1}^{n} \nabla f_i(w_{t-1}), \tag{4.5}$$

where η_t is a step size chosen at every iteration. Gradient descent thus works by taking steps along the direction of steepest descent in the optimization landscape.

One issue with gradient descent is that the calculation of ∇F requires calculating n gradients ∇f_i, which is very expensive for large n. Hence, a popular stochastic modification

of GD is stochastic gradient descent (SGD) [61], also known as mini-batch gradient descent. It follows the following update rule for $t \in \mathbb{N}$,

$$w_t \;=\; w_{t-1} - \frac{\eta_t}{m} \sum_{i \in I_t} \nabla f_i(w_{t-1}), \tag{4.6}$$

where η_t is again a step size, while $I_t \subset \{1, 2, \ldots, n\}$ is a index set of fixed cardinality $m < n$ chosen randomly without replacement at iteration t. The main advantage of SGD over GD is the fact that only m gradients are being calculated at every iteration. In practice we typically take $m \ll n$, which makes every individual iteration a lot faster than that of gradient descent. The disadvantage, however, is that the variance introduced by the randomness of the chosen mini-batch slows down convergence near the minima. Smaller the value of m, higher the variance of the stochastic gradient $\frac{1}{m}\sum_{i \in I_t} \nabla f_i(w_{t-1})$. While in GD we get linear convergence if we set $\eta_t = \eta$ as long as η is sufficiently small, in SGD we need to pick $\eta_t = O(\frac{1}{t})$ and even that gives us a sub-linear convergence rate of $O(\frac{1}{t})$. Inspite of this disadvantage, SGD enjoys widespread use in ERM today especially in situations where we want to come reasonably close to the minima, but do not care about its exact value.

4.3.2 Stochastic variance reduced gradient (SVRG) method

Stochastic variance reduced gradient (SVRG) method [31] is a hybrid technique leveraging the characteristics of both GD as well as SGD, which reduces the variance of the the parameter updates at every epoch and hence achieves faster convergence. In the start of the sth epoch in SVRG, the full gradient ∇F is computed at the snapshot $\tilde{w}_{s-1}$, which is updated in every epoch. The SVRG gradient estimator at iteration t of epoch s is defined as

$$g_t = \frac{1}{b}\sum_{i \in I_t} \nabla f_i(w_{t-1}) - \frac{1}{b}\sum_{i \in I_t} \nabla f_i(\tilde{w}_s) + \nabla F(\tilde{w}_{s-1}), \tag{4.7}$$

which has a lower variance than the gradient estimator in case of SGD, i.e. $\frac{1}{b}\sum_{i \in I_t} \nabla f_i(w_{t-1})$. Unlike SGD, for SVRG the step size η_t is not required to decay in order to achieve convergence. This leads to faster convergence, which was shown to be linear in [31]. Our implementation of SVRG in this chapter is one of its variants called VR-SGD proposed in [65], which we then modified to work on mini-batches instead of a single datum in every iteration. The procedure is described in Algorithm 1.

Algorithm 1 SVRG

Input: Data, initial $\tilde{w}_0$, number of epochs S, number of iterations m every epoch, mini-batch size b, step size η.

for $s = 1, 2, \ldots, S$ **do**
  $\tilde{\mu}_{s-1} = \nabla F(\tilde{w}_{s-1}) = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(\tilde{w}_{s-1})$
  $w_0 = \tilde{w}_{s-1}$
  for $t = 1, 2, \ldots, m$ **do**
    Pick $I_t \subset \{1, 2, \ldots, n\}$ randomly without replacement, with $|I_t| = b$.
    $g_t = \frac{1}{b}\sum_{i \in I_t} \nabla f_i(w_{t-1}) - \frac{1}{b}\sum_{i \in I_t} \nabla f_i(\tilde{w}_{s-1}) + \tilde{\mu}_{s-1}$
    $w_t = w_{t-1} - \eta g_t$
  end for
  $\tilde{w}_s = \frac{1}{m}\sum_{t=1}^{m} w_t$
end for

Output: $\tilde{w}_S$

We specifically chose VR-SGD not only due to its superior performance over the original implementation of SVRG, but also because the objective function converges to the minima in a very smooth linear manner every epoch. This makes it perform better empirically when accelerated via incorporating second-order information.

4.3.3 Subsampled Newton acceleration of SVRG

Second-order methods such as the Newton method incorporate curvature information, and significantly improves convergence of ERM algorithms. However, calculating and inverting the Hessian matrix involves high computational cost and memory. Thus, subsampling the Hessian is often preferred to ease computational burden. The challenge is to incorporate second-order information for acceleration in stochastic setting.

SVRG+I [34] focuses on incorporating an approximation of the curvature information using subsampled hessian, to accelerate SVRG. [21] showed that subsampled Hessian can capture accurate information only along high curvature directions. Since the estimation of curvature in the low curvature directions is very inaccurate, they proposed thresholding the low singular values to stabilize the estimate, resulting in conservative steps in the estimated low curvature directions.

The subsampling and thresholding of Hessian at point w in SVRG+I works as follows: for a sample $I \subset \{1, 2, \dots n\}$ with $|I| = B$ chosen randomly without replacement, calculate the subsampled hessian $\frac{1}{B}\sum_{i\in I} \nabla^2 f_i(w)$, whose singular values we shall denote by $\sigma_1, \dots, \sigma_d$ in descending order. For a given r, let the diagonal matrix $\Sigma_r \in \mathbb{R}^{r\times r}$ contain the top r singular singular values of the subsampled hessian, while $Q_r \in \mathbb{R}^{d\times r}$ contains the corresponding singular vectors. Then the estimator for the inverse of the Hessian at w after thresholding, is given by

$$\widehat{H_{s-1}^{-1}} \;=\; Q_r \left(\Sigma_r^{-1} - \frac{1}{\sigma_{r+1} I_r}\right) Q_r' \;+\; \frac{1}{\sigma_{r+1}} I_d, \tag{4.8}$$

This makes $\widehat{H_{s-1}^{-1}}$ the subsampled Hessian whose singular values from σ_{r+2} to σ_d are set to be equal to σ_{r+1}.

The entire procedure is described in Algorithm 2. In the beginning of every epoch, the

subsampled hessian is calculated at the snapshot $\tilde{w}_{s-1}$ using (4.8), and in every iteration of that particular epoch, the parameter estimates are updated by descending along the direction of $-\widehat{H_{s-1}^{-1}} g_t$, where g_t is the SVRG gradient estimator from (4.7). Once again, we have made minor changes in our implementation by assigning the average of the parameter updates over an entire epoch as the update to the snapshot of the next epoch, as in VR-SGD.

Algorithm 2 SVRG+I

Input: Data, initial $\tilde{w}_0$, number of epochs S, number of iterations m every epoch, mini-batch size b, batch size B for subsampling Hessian, number of singular values r for thresholding, step size η.

for $s = 1, 2, \ldots, S$ **do**
  $\tilde{\mu}_{s-1} = \nabla F(\tilde{w}_{s-1}) = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(\tilde{w}_{s-1})$
  Pick $I_s^H \subset \{1, 2, \ldots, n\}$ randomly without replacement, with $|I_s^H| = B$.
  Calculate $\widehat{H_{s-1}^{-1}}(\tilde{w}_{s-1})$ and threshold using (4.8) corresponding to I_s^H and r.
  $w_0 = \tilde{w}_{s-1}$
  for $t = 1, 2, \ldots, m$ **do**
    Pick $I_t \subset \{1, 2, \ldots, n\}$ randomly without replacement, with $|I_t| = b$.
    $g_t = \frac{1}{b} \sum_{i \in I_t} \nabla f_i(w_{t-1}) - \frac{1}{b} \sum_{i \in I_t} \nabla f_i(\tilde{w}_{s-1}) + \tilde{\mu}_{s-1}$
    $w_t = w_{t-1} - \eta \widehat{H_{s-1}^{-1}} g_t$
  end for
  $\tilde{w}_s = \frac{1}{m} \sum_{t=1}^{m} w_t$
end for

Output: $\tilde{w}_S$

4.3.4 Quasi-Newton acceleration of SVRG

Instead of subsampled Hessian, SVRG+II [34] uses the popular LBFGS method (see [48]) for the Hessian approximation to accelerate SVRG. The advantage of SVRG+II is that not only do we not have to compute a subsampled Hessian, we do not have to invert it either.

The estimate for the inverse Hessian, $\widehat{H}_{s-1}$, is calculated in epoch s using the curvature

information pair $(\Delta\tilde{w}_{s-2}, \Delta\tilde{\mu}_{s-2})$, where

$$\begin{aligned}\Delta\tilde{w}_{s-2} &= \tilde{w}_{s-1} - \tilde{w}_{s-2},\\ \Delta\tilde{\mu}_{s-2} &= \tilde{\mu}_{s-1} - \tilde{\mu}_{s-2},\end{aligned} \tag{4.9}$$

respectively encapsulate the epoch-to-epoch change in the snapshot $\tilde{w}$, as well the full gradient computed at the same. Then, $\widehat{H}_{s-1}$ is updated via the popular BFGS formula

$$\widehat{H}_{s-1} = \left(I - \frac{\Delta\tilde{w}_{s-2}\Delta\tilde{\mu}_{s-2}^T}{\Delta\tilde{w}_{s-2}^T\Delta\tilde{\mu}_{s-2}}\right)\widehat{H}_{s-2}\left(I - \frac{\Delta\tilde{\mu}_{s-2}\Delta\tilde{w}_{s-2}^T}{\Delta\tilde{\mu}_{s-2}^T\Delta\tilde{w}_{s-2}}\right) + \frac{\Delta\tilde{w}_{s-2}\Delta\tilde{w}_{s-2}^T}{\Delta\tilde{\mu}_{s-2}^T\Delta\tilde{w}_{s-2}} \tag{4.10}$$

The parameter estimates are updated every iteration in epoch s by descending along the direction $-\widehat{H}_{s-1}g_t$, where g_t is the SVRG gradient estimator from (4.7). The entire procedure is described in Algorithm 3.

Algorithm 3 SVRG+II

Input: Data, initial $\tilde{w}_0$, number of epochs S, number of iterations m every epoch, mini-batch size b, step size η.

Run one epoch of SVRG to obtain $\tilde{w}_1$ (and by extension, $\tilde{\mu}_0$).
Initialize $\widehat{H}_0$ to be the identity matrix.
for $s = 2, 3, \ldots, S$ **do**
    $\tilde{\mu}_{s-1} = \nabla F(\tilde{w}_{s-1}) = \frac{1}{n}\sum_{i=1}^n \nabla f_i(\tilde{w}_{s-1})$
    $\Delta\tilde{w}_{s-2} = \tilde{w}_{s-1} - \tilde{w}_{s-2}$
    $\Delta\tilde{\mu}_{s-2} = \tilde{\mu}_{s-1} - \tilde{\mu}_{s-2}$
    $\widehat{H}_{s-1} = \left(I - \frac{\Delta\tilde{w}_{s-2}\Delta\tilde{\mu}_{s-2}^T}{\Delta\tilde{w}_{s-2}^T\Delta\tilde{\mu}_{s-2}}\right)\widehat{H}_{s-2}\left(I - \frac{\Delta\tilde{\mu}_{s-2}\Delta\tilde{w}_{s-2}^T}{\Delta\tilde{\mu}_{s-2}^T\Delta\tilde{w}_{s-2}}\right) + \frac{\Delta\tilde{w}_{s-2}\Delta\tilde{w}_{s-2}^T}{\Delta\tilde{\mu}_{s-2}^T\Delta\tilde{w}_{s-2}}$
    $w_0 = \tilde{w}_{s-1}$
    for $t = 1, 2, \ldots, m$ **do**
        Pick $I_t \subset \{1, 2, \ldots, n\}$ randomly without replacement, with $|I_t| = b$.
        $g_t = \frac{1}{b}\sum_{i\in I_t}\nabla f_i(w_{t-1}) - \frac{1}{b}\sum_{i\in I_t}\nabla f_i(\tilde{w}_s) + \tilde{\mu}_{s-1}$
        $w_t = w_{t-1} - \eta\widehat{H}_{s-1}g_t$
    end for
    $\tilde{w}_s = \frac{1}{m}\sum_{t=1}^m w_t$
end for

Output: $\tilde{w}_S$

4.4 Expectation Maximization (EM)

The expectation maximization algorithm is designed for models with some observed variable x and unobserved latent variable z. The setting is the following. Suppose a dataset of n observations $X = \{x_i\}_{1 \leq i \leq n}$ is provided to us, while the corresponding hidden variables $Z = \{z_i\}_{1 \leq i \leq n}$ are unobserved. The pairs (x_i, z_i) are assumed to be i.i.d.. Our goal then, is to find the maximum likelihood estimate of the parameter $w \in \mathbb{R}^d$, by maximizing the marginal log-likelihood

$$F(w) = \sum_{i=1}^{n} \log p(x_i; w) = \sum_{i=1}^{n} \log \int_{h_i} p(x_i, h_i; w) dh_i,$$

which is often considered intractable.

4.4.1 EM

The EM algorithm finds the MLE of the marginal log-likelihood by iteratively applying the two steps:

- Expectation step: Define

$$Q(w|w_t) = \mathbb{E}_{Z \sim p(.|X, w_t)} \left(\log p(X, Z|w) \right), \tag{4.11}$$

  i.e., let $Q(w|w_t)$ be the expected value of the log-likelihood with respect to the current conditional distribution of Z given X and the current parameter estimate w_t.

- Maximization step: $w_{t+1} = \arg\max_w Q(w|w_t)$.

These two steps can be combined into one function and presented in a concise manner.

Define

$$\Psi(w) = \arg\max_{w} \mathbb{E}_{Z\sim p(.|X,w_t)} \left(\log p(X, Z|w)\right), \tag{4.12}$$

then, the EM update at parameter w becomes

$$w_{t+1} = \Psi(w_t). \tag{4.13}$$

The EM algorithm typically converges linearly [42]. Unlike gradient descent and its variants, EM algorithm does not have a tunable hyperparameter like step size. Analogous to SGD, stochastic EM (sEM) was proposed in [15] which possesses both the pros and cons of SGD. In iteration t of the mini-batch variant of sEM, an index set $I_t \subset \{1, 2, \ldots, n\}$ of fixed cardinality $m \ll n$ is sampled randomly without replacement, and the parameter is updated via

$$w_{t+1} = \psi_{I_t}(w_t), \tag{4.14}$$

where

$$\psi_{I_t}(w) = \arg\max_{w} \mathbb{E}_{Z\sim p(.|\{x_i\}_{i\in I_t},w_t)} \left(\log p(\{x_i\}_{i\in I_t}, \{z_i\}_{i\in I_t}|w_t)\right). \tag{4.15}$$

It enjoys much faster initial convergence due to cheap updates, but the convergence rate, like SGD, falls to at best $O(\frac{1}{t})$ near the minima with a decaying step size [15].

4.4.2 Variance reduced stochastic expectation maximization (sEM-vr) method

The variance reduced stochastic expectation maximization (sEM-vr) algorithm was introduced in [16] to leverage advantages of both EM and sEM, and has a lot of similarities with SVRG. The main similarity is that the EM step $w_t - w_{t+1}$ can be viewed as an analogue

of the gradient in ERM. In particular, for an index set I_t of cardinality b, $w_t - \psi_{I_t}(w_t)$ in sEM is an analogue for $\frac{1}{b}\sum_{i\in I_t}\nabla f_i(w_t)$ in SGD. Similarly, $w_t - \Psi(w_t)$ in EM is an analogue for $\nabla F(w_t)$ in GD. Thus, a variance reduced gradient analogue of (4.7) for expectation maximization, would be

$$(w_{t-1} - \psi_{I_t}(w_{t-1})) - (\tilde{w}_{s-1} - \psi_{I_t}(\tilde{w}_{s-1})) + (\tilde{w}_{s-1} - \Psi(\tilde{w}_{s-1}))$$

which simplifies to

$$(w_{t-1} - \psi_{I_t}(w_{t-1})) + \psi_{I_t}(\tilde{w}_{s-1}) - \Psi(\tilde{w}_{s-1}) \tag{4.16}$$

The sEM-vr update in (4.16) can be shown to have a lower variance than that of sEM. The main procedure of sEM-vr closely follows that of SVRG. At the start of the sth epoch, the EM update on the full data is computed at the snapshot $\tilde{w}_{s-1}$, which is updated in every epoch. In iteration t of epoch s, the parameter w_t is updated by descending opposite to the direction of (4.16) with a step size of η. The entire procedure is described in Algorithm 4.

Algorithm 4 sEM-vr

Input: Data, initial $\tilde{w}_0$, number of epochs S, number of iterations m every epoch, mini-batch size b, step size η.

for $s = 1, 2, \ldots, S$ **do**
  Compute and save $\Psi(\tilde{w}_{s-1})$ for current epoch.
  $w_0 = \tilde{w}_{s-1}$
  for $t = 1, 2, \ldots, m$ **do**
    Pick $I_t \subset \{1, 2, \ldots, n\}$ randomly without replacement, with $|I_t| = b$.
    $w_t = w_{t-1} - \eta\left((w_{t-1} - \psi_{I_t}(w_{t-1})) + \psi_{I_t}(\tilde{w}_{s-1}) - \Psi(\tilde{w}_{s-1})\right)$
  end for
  $\tilde{w}_s = \frac{1}{m}\sum_{t=1}^{m} w_t$
end for

Output: $\tilde{w}_S$

4.4.3 Quasi-Newton acceleration of sEM-vr

A subsampled Hessian-based acceleration of sEM-vr is not directly possible, since the EM update can have a jacobian matrix which may not even be symmetric. However, a quasi-Newton acceleration of sEM-vr is indeed possible, following the same spirit as that of SVRG+II. Using the analogues of gradient in EM case as mentioned in Section 4.4.2, SVRG+II can be naturally modified to act as a quasi-Newton acceleration of sEM-vr, which we shall refer to as sEM-vr+QN. To the best of our knowledge, this is an original contribution.

Defining $\tilde{\mu}_{s-1} := \Psi(\tilde{w}_{s-1})$, the estimate for the inverse Hessian, $\widehat{H}_{s-1}$, is calculated in epoch s using the curvature information pair $(\Delta\tilde{w}_{s-2}, \Delta\tilde{\mu}_{s-2})$, as defined in (4.9). $\widehat{H}_{s-1}$ is then updated via the BFGS formula (4.10). Parameter estimates are updated every iteration in epoch s by descending along the direction $-\widehat{H}_{s-1}g_t$, where g_t is the sEM-vr update direction from (4.16). The sEM-vr+QN procedure is described in Algorithm 5.

Algorithm 5 sEM-vr+QN

Input: Data, initial $\tilde{w}_0$, number of epochs S, number of iterations m every epoch, mini-batch size b, step size η.

Run one epoch of sEM-vr to obtain $\tilde{w}_1$ (and by extension, $\tilde{\mu}_0 := \Psi(\tilde{w}_0)$).
Initialize $\widehat{H}_0$ to be the identity matrix.
for $s = 2, 3, \ldots, S$ **do**
    Compute and save $\tilde{\mu}_{s-1} := \Psi(\tilde{w}_{s-1})$ for current epoch.
    $\Delta\tilde{w}_{s-2} = \tilde{w}_{s-1} - \tilde{w}_{s-2}$
    $\Delta\tilde{\mu}_{s-2} = \tilde{\mu}_{s-1} - \tilde{\mu}_{s-2}$
    $\widehat{H}_{s-1} = \left(I - \frac{\Delta\tilde{w}_{s-2}\Delta\tilde{\mu}_{s-2}^T}{\Delta\tilde{w}_{s-2}^T\Delta\tilde{\mu}_{s-2}}\right)\widehat{H}_{s-2}\left(I - \frac{\Delta\tilde{\mu}_{s-2}\Delta\tilde{w}_{s-2}^T}{\Delta\tilde{\mu}_{s-2}^T\Delta\tilde{w}_{s-2}}\right) + \frac{\Delta\tilde{w}_{s-2}\Delta\tilde{w}_{s-2}^T}{\Delta\tilde{\mu}_{s-2}^T\Delta\tilde{w}_{s-2}}$
    $w_0 = \tilde{w}_{s-1}$
    for $t = 1, 2, \ldots, m$ **do**
        Pick $I_t \subset \{1, 2, \ldots, n\}$ randomly without replacement, with $|I_t| = b$.
        $w_t = w_{t-1} - \eta\widehat{H}_{s-1}\left((w_{t-1} - \psi_{I_t}(w_{t-1})) + \psi_{I_t}(\tilde{w}_{s-1}) - \tilde{\mu}_{s-1}\right)$
    end for
    $\tilde{w}_s = \frac{1}{m}\sum_{t=1}^{m} w_t$
end for

Output: $\tilde{w}_S$

4.5 Numerical experiments

Notice that from (1.3), the negative log-likelihood of an unknown function v is given by

$$\begin{aligned} F(v) &= -\frac{1}{N}\sum_{j=1}^{N}\log\int_{g\in G}\exp\left\{-\frac{1}{2\sigma^2}\left\|\mathcal{A}(g_j\circ v)-v_j\right\|^2\right\}\rho(g), \\ &= \frac{1}{N}\sum_{j=1}^{N} f_j(v), \end{aligned} \tag{4.17}$$

where f_j is the negative log-likelihood corresponding to a single observation v_j. Clearly, this fits into the ERM setting described in the beginning of Section 4.3. Also, observe that the N observations v_j correspond to unobserved latent variables g_j. This also makes orbit recovery problems fit into the EM setting described in the beginning of Section 4.4. In the next two subsections, we conduct numerical experiments for windowed MRA and cryo-EM case, where we compare the various methods for ERM and EM.

4.5.1 Windowed multireference alignment

Let $\widehat{v}\in\mathbb{C}^{\tilde{n}}$ be a given observation from the windowed multireference alignment model. The negative log-likelihood of $w=[\widehat{x},\rho]^T$ is given by

$$f(w) = -\log\sum_{l=0}^{n-1}\exp\left\{\log\rho[l]-\frac{1}{2\sigma^2}\left\|\tilde{R}_l\widehat{x}-\widehat{v}\right\|^2\right\}, \tag{4.18}$$

where $\widehat{x}\in\mathbb{C}^n$ is a signal, and $\rho\in\mathbb{R}^n$ is a density vector of shifts, i.e. the probability mass function. Clearly $\sum_{i=1}^{N}\rho[i]=1$, i.e. ρ lies on the positive orthant of the N-dimensional 1-norm unit sphere S_1^N. However, in all the ERM methods, after an update in any iteration, there is no guarantee that the updated ρ will also be an element of S_1^N. Hence we

reparameterize ρ by

$$\rho = \text{softmax}(\tilde{\rho})$$

for some $\tilde{\rho} \in \mathbb{R}^N$.

The gradient of f with respect to $\widehat{x}$ is given by

$$\nabla_{\widehat{x}} f(\widehat{x}) \;=\; \frac{1}{\sigma^2} \begin{bmatrix} \tilde{R}_0^*(\tilde{R}_0 \widehat{x} - \widehat{v}) \\ \vdots \end{bmatrix} \text{softmax} \left(\begin{bmatrix} \log(\text{softmax}(\tilde{\rho}))[0] - \frac{1}{2\sigma^2} \left\| \tilde{R}_0 x - \widehat{v} \right\|^2 \\ \vdots \end{bmatrix} \right),$$

while that with respect to $\tilde{\rho}$ is given by

$$\begin{aligned} \nabla_{\tilde{\rho}} f(\tilde{\rho}) \;&= (diag(\tilde{\rho}) - \tilde{\rho}\tilde{\rho}^T)\; diag(\text{softmax}(\tilde{\rho}))^{-1} \\ &\times\; \text{softmax} \left(\begin{bmatrix} \log(\text{softmax}(\tilde{\rho}))[0] - \frac{1}{2\sigma^2} \left\| \tilde{R}_0 x - \widehat{v} \right\|^2 \\ \vdots \end{bmatrix} \right). \end{aligned}$$

Using f and ∇f, we can calculate the updates of gradient descent, SVRG, and SVRG+II for windowed MRA. The hessian for SVRG+I can be calculated using autograd.

Similar calculations can be done for the EM algorithm in case of windowed MRA. The log posterior (up to a constant) is given by

$$\log p\left(\{\widehat{v}_j\}_{j \le N}, \{\widehat{s}_j\}_{j \le N} | w \right) \;=\; \sum_{j=1}^{N} \left[\log \text{softmax}(\tilde{\rho})[s_j] - \frac{1}{2\sigma^2} \left\| \tilde{R}_{s_j} \widehat{x} - \widehat{v}_j \right\|^2 \right],$$

where $w = [\widehat{x}, \tilde{\rho}]^T$, and s_j are the hidden variables, i.e. the shifts corresponding to the observations $\widehat{x}_j$.

As per (4.12), the EM update is given by

$$\Psi(w_t) = \arg\max_{w} \mathbb{E}_{s\sim p(.|\{\widehat{v}_j\},w_t)} \left(\log p(\{\widehat{v}_j\},\{\widehat{s}_j\}|w)\right),$$
$$= \arg\max_{\widehat{x},\ \tilde{\rho}} \sum_{j=1}^{N}\sum_{l=0}^{n-1} \omega_t^{l,j} \left[\log \text{softmax}(\tilde{\rho})[l] - \frac{1}{2\sigma^2}\left\|\tilde{R}_l\widehat{x} - \widehat{v}_j\right\|^2\right], \tag{4.19}$$

where

$$\omega_t^{l,j} = P\left(s_j = \frac{2\pi l}{n} \;\middle|\; \{\widehat{v}_j\}, w_t\right)$$
$$= \text{softmax}\left(\log \text{softmax}(\tilde{\rho}_t)[l] - \frac{1}{2\sigma^2}\left\|\tilde{R}_l\widehat{x}_t - \widehat{v}_j\right\|^2\right). \tag{4.20}$$

Solving for the argmax in (4.19) gives us that

$$\widehat{x}_{t+1} = \left(\sum_{j=1}^{N}\sum_{l=0}^{n-1} \omega_t^{l,j}\tilde{R}_l^*\tilde{R}_l\right)^{-1}\left(\sum_{j=1}^{N}\sum_{l=0}^{n-1} \omega_t^{l,j}\tilde{R}_l^*\widehat{v}_j\right),$$
$$\tilde{\rho}_{t+1}[l] = \log\left[\frac{\sum_{j=1}^{N}\omega_t^{l,j}}{\sum_{j=1}^{N}\sum_{k=0}^{n-1}\omega_t^{k,j}}\right], \qquad \forall\ 0 \le l \le n-1.$$

Notice that $\omega_t^{l,j}$ is a function of $\widehat{x}_t$ and $\tilde{\rho}_t$. The aforementioned EM updates for windowed MRA are also used in sEM-vr and sEM-vr+QN.

For our experiments with windowed MRA, we took a Gaussian pulse with $N = 16$ as the ground truth signal, and rescaled it to have norm 1. The window length is taken to be 12, while the distribution of shifts were taken to be a mixture of two Gaussians. 3000 observations are taken from this model and Gaussian noise with $\sigma = 0.2$ is added to them. The signal and histogram of shifts can be seen in Figure 4.1. The convergence plots of our discussed methods for ERM are plotted in Figure 4.2, while those for EM and its variants are plotted in Figure 4.3.

$|F - F_*|$ and $|\Psi - \Psi_*|$ are plotted along the y-axes of these convergence plots in common

logarithmic scale, for ERM and EM respectively. The global minimas F_* (respectively Ψ_*) is obtained by running SVRG+II (respectively sEM-vr+QN) until convergence. Along the x-axis, we plot the number of cost-weighted epochs and wall-clock time in seconds. The cost-weighting of epochs is done on the basis of the number of effective passes through the entire dataset. For example, each epoch of GD requires 1 pass through the dataset, while for SVRG, the number of effective passes is $1 + \frac{mb}{N}$, where m and b are the number of iterations per epoch and the batch size respectively. SVRG+II requires the same number of effective passes as SVRG, while for SVRG+I it is $1 + \frac{nB+mb}{N}$, with n and B being the dimension of the parameter and the batch size for subsampling the Hessian respectively. The cost per epoch can be calculated accordingly for variants of EM. After reweighting according to cost, the number of cost-weighted epochs and wall-clock time till convergence (or until stopping criterion is reached) is also recorded and displayed in Tables 4.1 and 4.2 for ERM and EM respectively, where a method is said to have converged if the objective at 2 consecutive epochs differ by less than 10^{-14}.

We observe from Figure 4.2 and Table 4.1 that for ERM, SVRG has a visibly faster rate of convergence than GD. However, SVRG+I does not perform any acceleration over SVRG, instead, is actually slower. None of them converged until maximum stipulated number of cost-weighted iterations (i.e. 14000) were reached. SVRG+II was the fastest, and converged in 821 cost-weighted epochs. From Figure 4.3 and Table 4.2 that for EM, we see that similar pattern holds for EM. While all three of the tested methods converge in a reasonably short time in comparison to ERM, sEM-vr+QN and sEM-vr still outperform the usual EM algorithm.

4.5.2 Cryo-EM

For the cryo-em scenario, we shall limit ourselves to ERM and assume that the true density of rotations is known. Let $\widehat{v}_0(K_2) \in \mathbb{C}^{n\times n}$ be a given cryo-EM image, i.e. observation, with

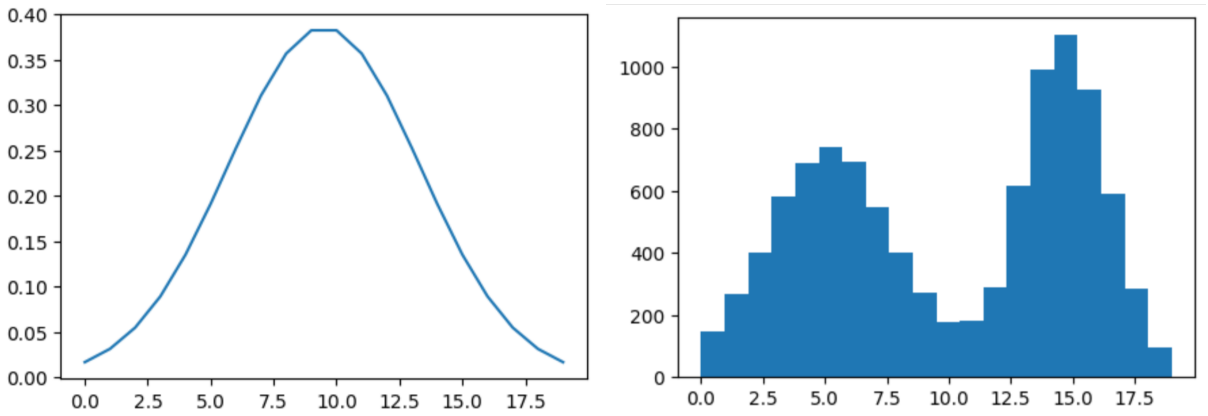

Figure 4.1: Plots of the signal to be estimated (left) and known distribution of shifts (right).

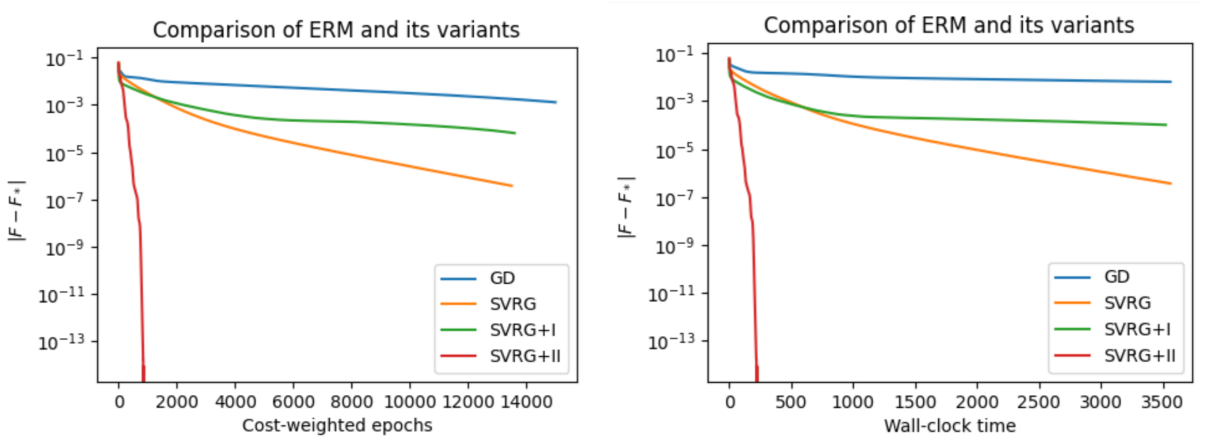

Figure 4.2: Convergence plots of $|F - F_*|$ with respect to cost-weighted epochs (left) and wall-clock time (right), for ERM and its variants in case of windowed multireference alignment of the signal and distribution of shifts shown in Figure 4.1.

Methods	Cost-weighted epochs	Wall-clock time
GD	14000+	3500+
SVRG	14000+	3500+
SVRG+I	14000+	3500+
SVRG+II	821	217

Table 4.1: Number of cost-weighted epochs and wall-clock time (in seconds) till convergence, for ERM methods in case of windowed multireference alignment of the signal and distribution of shifts shown in Figure 4.1.

K_2 being a grid of n^2 equispaced points on $[-\pi, \pi]^2$. Then, following (4.17), the negative log-likelihood of the volume $\widehat{v}$ is given by

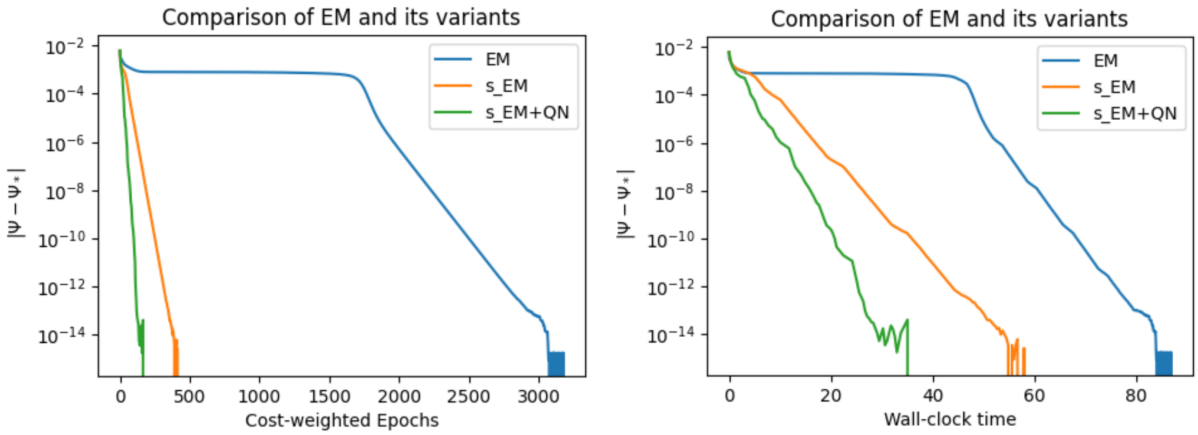

Figure 4.3: Convergence plots of $|\Psi - \Psi_*|$ with respect to cost-weighted epochs (left) and wall-clock time (right), for EM and its variants in case of windowed multireference alignment of the signal and distribution of shifts shown in Figure 4.1.

Methods	Cost-weighted epochs	Wall-clock time
EM	3078	85
sEM-vr	392	53
sEM-vr+QN	188	34

Table 4.2: Number of cost-weighted epochs and wall-clock time (in seconds) till convergence, for EM-based methods in case of windowed multireference alignment of the signal and distribution of shifts shown in Figure 4.1.

$$f(\widehat{v}) \;=\; -\log \int_{x\in \mathrm{SO}(3)} \exp\left\{-\frac{1}{2\sigma^2}\left\|S\circ R_j\circ \widehat{v}(K_2)-\widehat{v}_0(K_2)\right\|^2\right\}\rho(x), \tag{4.21}$$

where $\rho : \mathrm{SO}(3) \to [0,1]$ is a density function on the space of 3D rotations.

In this section, we aim to re-purpose our MRA calculations to fit the cryo-EM scenario. For that purpose, we have two challenges:

- We need to approximate the integration in (4.21) so that our windowed MRA calculations are applicable with minimal changes.

- We need a good way to parameterize the volume $\widehat{v}$ such that the slices $S \circ R_j \circ \widehat{v}(K_2)$

can be calculated easily.

To address the first point, we approximate the integral in (4.21) using the quadrature rule. We use the two-step process underlined in Section 2.4.2 for selecting a good set $Q \subset \mathrm{SO}(3)$ on which we use a quadrature rule with uniform weights. First we chose a q_1-point spherical design (see [75]) on S^2. Next, for every point of the design, treating the axis connecting the center to that point as a viewing direction, we take in-plane rotations with q_2 equispaced angles in $[0, 2\pi)$ radians. We thus obtained a set Q with $|Q| = q_1 q_2$ quadrature points on SO(3). For our purposes, let us denote the quadrature points as $\{R_k^{quad}\}_{1 \leq k \leq Q}$. Thus, the discrete approximation of (4.21) becomes

$$f(\widehat{v}) \;=\; -\log \sum_{j=1}^{Q} \exp\left\{ -\frac{1}{2\sigma^2} \left\| S \circ R_j^{quad} \circ \widehat{v}(K_2) - \widehat{v}_0(K_2) \right\|^2 \right\} \rho\left(R_j^{quad}\right). \tag{4.22}$$

As for the second point, it is convenient to represent the Fourier volume using a steerable basis, i.e. a function space that is closed under rotations. A popular such way is by using a combination of spherical harmonics and the spherical Bessel basis (see [66, 9]), as described in Section 3.3. Recall that a band-limited Fourier volume $\widehat{v}$ can be expanded to degree L as

$$\widehat{v}(k, \theta, \phi) \;\approx\; \sum_{l=0}^{L} \sum_{m=-l}^{l} \sum_{s=1}^{S(l)} A_{l,m,s} F_{l,s}(k) Y_l^m(\theta, \phi), \tag{4.23}$$

where k is the radial frequency, Y_l^m are complex spherical harmonics, and $F_{l,s}$ are the spherical Bessel functions [4]. Since our goal is low-resolution modeling, we can choose to limit L and S_l in order to reduce computational requirements. $A_{l,m,s}$ are therefore, the parameters we need to estimate in order to recover the volume.

Enumerating the three summations in (4.23) as a single sum, we can rewrite it as

$$\widehat{v}(k_x, k_y, k_z) \;\approx\; \sum_{p=0}^{\tilde{L}} w_p \; \tilde{Y}_p(k_x, k_y, k_z), \qquad (k_x, k_y, k_z) \in [-\pi, \pi]^3, \tag{4.24}$$

where w_p and $\tilde{Y}_p(k_x, k_y, k_z)$ correspond to a specific $A_{l,m,s}$ and $F_{l,s}(k)Y_l^m(\theta, \phi)$ respectively, with (k, θ, ϕ) being the spherical coordinate representation of (k_x, k_y, k_z). Therefore we have from (4.22), that for all $j \in \{1, 2, \ldots, Q\}$,

$$\begin{aligned} S \circ R_j^{quad} \circ \widehat{v}(K_2) \;&\approx\; \sum_{p=0}^{\tilde{L}} w_p \; S \circ R_j^{quad} \circ \tilde{Y}_p(K_2) \\ &:=\; M_j w, \end{aligned}$$

where $w = \left[w_0, w_1, \ldots, w_{\tilde{L}}\right]^T \in \mathbb{C}^{\tilde{L}}$ and $M_j \in \mathbb{C}^{n^2 \times \tilde{L}}$ is a matrix whose pth column is $S \circ R_j^{quad} \circ \tilde{Y}_p(K_2)$. Note that M_js are known matrices that are calculated and fixed before reconstruction, hence w is the only parameter representing the volume to be estimated. (4.22) therefore becomes

$$f(w) \;=\; -\log \sum_{j=1}^{Q} \exp\left\{\log \rho[j] - \frac{1}{2\sigma^2}\left\|M_j w - \widehat{v}_0(K_2)\right\|^2\right\}, \tag{4.25}$$

where by abuse of notation, $\rho \in \mathbb{C}^Q$ is a vector whose jth coordinate is $\rho\left(R_j^{quad}\right)$. This form is practically identical to that of (4.18) for windowed MRA, hence the gradient can be calculated similarly.

For our experiments, we take $q_1 = 100$ and $q_2 = 12$ for a total of $|Q| = 1200$ quadrature points. The parameters for spherical harmonic representation of the volume are taken as $L = 3$, $S(l) = 4$. Note that since our Fourier volume is the Fourier transform of a real volume,

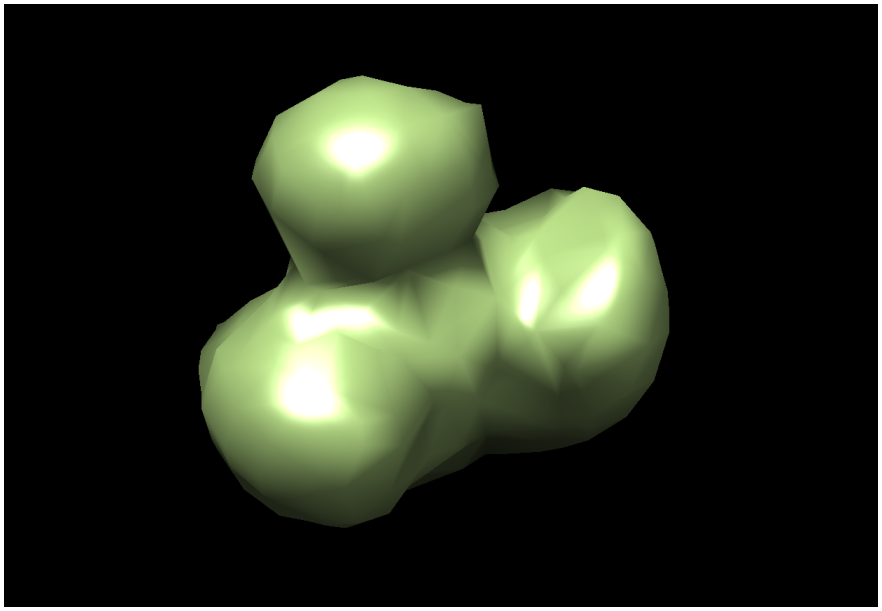

Figure 4.4: Low-dimensional representation of mixture of 6 Gaussians

it is conjugate symmetric. This puts a lot of restrictions on $A_{l,m,s}$ in (4.23), effectively reducing the dimensionality of the problem to $\tilde{L} = 40$. The volume we consider is a spherical harmonic representation of a mixture of 6 Gaussians. For this volume, we generate $N = 3600$ cryo-EM image observations of size 15×15 with respect to uniform distribution on SO(3), with an SNR of approximately 0.34. The volume is depicted in Figure 4.4, while two sample observations (one clean and one noisy) are displayed in Figure 4.5. The convergence plots of our discussed methods for ERM (except SVRG+I) are plotted in Figure 4.6. We have omitted SVRG+I since its performance is not competitive with the unaccelerated SVRG, as observed in case of MRA.

We observe from Figure 4.2 and Table 4.1 that for ERM, SVRG and SVRG+II again outperform GD, with SVRG+II being the fastest.

4.6 Conclusion

In this chapter, we have experimented with the application and development of variance-reduced methods for accelerating variance-reduced EM and ERM in the context of orbit

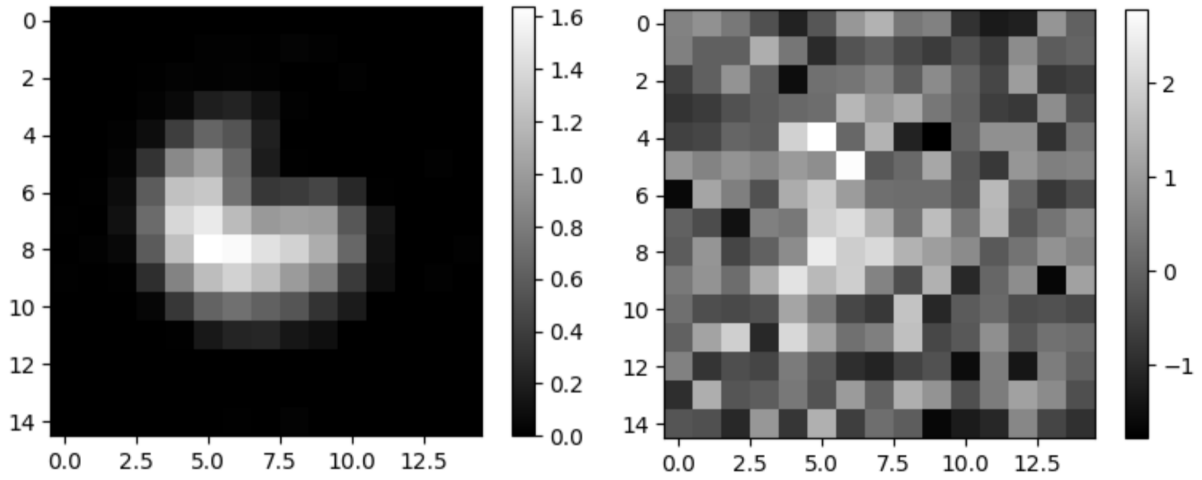

Figure 4.5: A clean (left) and a noisy sample observation (right), for testing accelerated ERM methods.

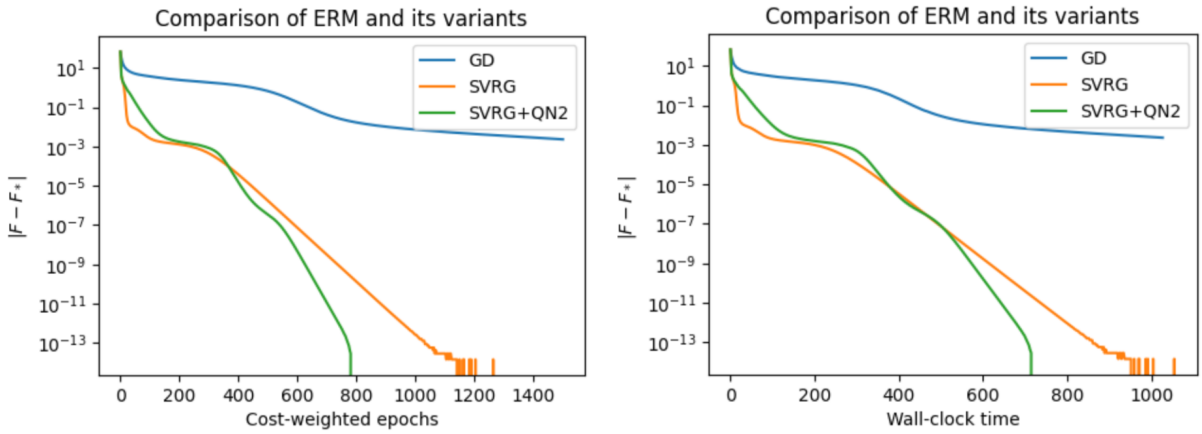

Figure 4.6: Convergence plots of $|F - F_*|$ with respect to cost-weighted epochs (left) and wall-clock time (right), for ERM and its variants in case of cryo-EM reconstruction of the volume in Figure 4.4.

Methods	Cost-weighted epochs	Wall-clock time
GD	2000+	1500+
SVRG	1139	942
SVRG+II	790	713

Table 4.3: Number of cost-weighted epochs and wall-clock time (in seconds) till convergence, for ERM-based methods in case of cryo-EM reconstruction of the volume shown in Figure 4.4.

recovery problems, leveraging second-order information. Even first-order variance-reduced (VR) methods display evident superior performance in comparison to traditional ERM and EM algorithms. By incorporating curvature information, our methods are shown to often achieve further computational efficiency and enhanced convergence rates compared to traditional first-order VR and non-VR methods.

Our experimental results on simulated datasets of two classes of orbit recovery problems, namely multireference alignment and cryo-EM, is a proof-of-concept demonstration that these variance-reduced algorithms of both first as well as second-order outperform their unaccelerated counterparts. Further research is required to establish superiority of these methods in high-dimensional settings typical of cryo-EM. We have also proposed a method for quasi-Newton acceleration of sEM-vr, which according to our experiments outperform the first-order variant in windowed MRA setting.

In conclusion, the integration of second-order information and variance reduction in EM and ERM algorithms represents a significant advancement in the field of statistical estimation. The promising results obtained in orbit recovery settings suggest that these methods can be effectively applied to a wide range of other inverse problems, which could be also be explored in future work. This would offer a robust tool for researchers and practitioners seeking to enhance the performance of their analytical models. Additionally, further refinement of these algorithms could involve adaptive schemes that dynamically adjust the use of second-order information based on the problem's characteristics, potentially offering even greater efficiency and robustness.

CHAPTER 5

FINAL THOUGHTS

Single-particle cryo-EM is a prominent method for determining the atomic-resolution 3D structure of biological macromolecules. This technique underwent a "resolution revolution" a decade ago [36], and three of its pioneers were awarded the 2017 Nobel Prize in Chemistry [19]. Over the past few years, cryo-EM has provided researchers with access to some of the molecules' smallest and most essential building blocks. One of its simpler versions, known as the multireference alignment problem, is also of interest in areas like signal processing. Broadly speaking, our primary contribution this thesis is two-fold:

- We developed deep neural network-based frameworks for solving moment systems in orbit recovery problems, which both cryo-EM and multireference alignment fall under.
- We achieved faster convergence while solving orbit recovery problems via maximum likelihood estimation by applying variance reduced methods, which we then further accelerated by using second-order information.

For both multireference alignment as well as cryo-EM, we showed that a map can be learned to take moments of orbit recovery observations as input and output the underlying signal/volume and density of group elements. We then developed novel neural network architectures to learn this map in a supervised manner. This enables us to obtain us a near-instantaneous and cost-effective preliminary estimate of the underlying signal/volume. This map was then used as a deep neural network prior to accelerate convergence in unsupervised reconstruction when provided moments from a previously unseen dataset. The setting of cryo-EM that we dealt with, was a general one which allowed for small translations in the data images. We named the pipeline for dealing with this general cryo-EM setting MoM-net.

The biggest advantage of our neural network framework for MoM, is feasibility of supervised learning of the inversion map. If we had to directly learn the map from the set of cryo-

EM datasets to the set of underlying structures, it would have been completely intractable given the massive sizes of cryo-EM datasets. Any method aiming to learn this inversion map would have to find some statistics from a dataset that encapsulates enough information to recover the volume, otherwise supervised training would be impossible. Our framework uses moments to encapsulate the necessary information from datasets, hence making supervised training possible. Apart from this, the method of moments itself offers benefits like noise resilience and scalability, requiring only a single pass over the entire dataset. A big drawback of our framework is that it can only recover the underlying signals/volumes on the basis of the information captured in the low-order moments, which in case of cryo-EM, is often not enough to obtain a high-resolution reconstruction. Other issues include the ill-posed nature of the problem. While this issue is somewhat alleviated by using the "good" initialization provided by the neural network prior, the reconstruction process can still get stuck in spurious local minimas in the optimization landscape.

We then changed frameworks and explored the application of variance-reduced empirical risk minimization (ERM) and expectation maximization (EM) methods to carry out maximum likelihood estimation in the context of orbit recovery problems. We showed that the first-order variance-reduced (VR) counterparts of traditional ERM and EM algorithms exhibit superior performance in comparison to them. Incorporation of second-order information led to even further acceleration in convergence of the same. We also proposed a quasi-Newton acceleration method for stochastic variance-reduced expectation maximization (sEM-vr), which outperforms the first-order sEM-vr in windowed multireference alignment setting.

A potential next step for the work done in this thesis would be to develop a *complete* pipeline for high-resolution reconstruction in orbit recovery problems. To elaborate, when provided with an unknown dataset, our pipeline should be able to generate *ab-initio* reconstructions from moments of the data (expedited using a trained neural network) along the

lines of Chapters 2 and 3, and then further refine them using variance reduced algorithms to provide high resolution reconstructions of the underlying signal/volume, along the lines of Chapter 4. Several challenges would need to be addressed for this to come to fruition:

- Improving the resolution of the preliminary predicted volume via supervised learning would be desirable since this would lead to faster convergence during the ab-initio reconstruction process. Parallelizing the model on multiple GPUs, along with the usage of compressed moments during the training process could be explored, so that moments corresponding to larger sized images, or even higher order moments, can be used.

- Our experiments using variance-reduced methods mainly focused on low dimensional signals and volumes. However, if we were to use these methods to optimize the large number of parameters of MoM-net, it is certain that further experimentation is required in high-dimensional scenarios first to verify that our results hold.

Another avenue of research involves addressing conformational heterogeneity in the underlying volume for cryo-EM. Softwares such as RELION [44, 80] and cryoSPARC [56] are able to differentiate between various states and reconstruct each one with precision, but they do not exploit the advantages of the method of moments. Currently, MoM-net focuses on homogeneous reconstruction, but the capability to recover a limited number of conformations would be highly beneficial. Capturing the structural heterogeneity inherent in biological samples would significantly expand the practical applicability of MoM-net to real-world datasets.

REFERENCES

[1] Emmanuel Abbe, Tamir Bendory, William Leeb, João M Pereira, Nir Sharon, and Amit Singer. Multireference alignment is easier with an aperiodic translation distribution. *IEEE Transactions on Information Theory*, 65(6):3565–3584, 2018.

[2] Emmanuel Abbe, João M Pereira, and Amit Singer. Estimation in the group action channel. In *2018 IEEE International Symposium on Information Theory*, pages 561–565. IEEE, 2018.

[3] Joakim Andén and Amit Singer. Structural variability from noisy tomographic projections. *SIAM Journal on Imaging Sciences*, 11(2):1441–1492, 2018.

[4] Larry C Andrews. *Special functions of mathematics for engineers*, volume 49. Spie Press, 1998.

[5] Zhong-Zhi Bai, Wen-Ting Wu, and Galina V Muratova. The power method and beyond. *Applied Numerical Mathematics*, 164:29–42, 2021.

[6] Afonso S Bandeira, Ben Blum-Smith, Joe Kileel, Jonathan Niles-Weed, Amelia Perry, and Alexander S Wein. Estimation under group actions: recovering orbits from invariants. *Applied and Computational Harmonic Analysis*, 66:236–319, 2023.

[7] Alberto Bartesaghi, Alan Merk, Soojay Banerjee, Doreen Matthies, Xiongwu Wu, Jacqueline LS Milne, and Sriram Subramaniam. 2.2 Å resolution cryo-EM structure of β-galactosidase in complex with a cell-permeant inhibitor. *Science*, 348(6239):1147–1151, 2015.

[8] Tamir Bendory, Nicolas Boumal, Chao Ma, Zhizhen Zhao, and Amit Singer. Bispectrum inversion with application to multireference alignment. *IEEE Transactions on Signal Processing*, 66(4):1037–1050, 2017.

[9] Tamir Bendory, Yuehaw Khoo, Joe Kileel, Oscar Mickelin, and Amit Singer. Autocorrelation analysis for cryo-em with sparsity constraints: Improved sample complexity and projection-based algorithms. *Proceedings of the National Academy of Sciences*, 120(18):e2216507120, 2023.

[10] Tristan Bepler, Kotaro Kelley, Alex J Noble, and Bonnie Berger. Topaz-denoise: general deep denoising models for cryoEM and cryoET. *Nature communications*, 11(1):1–12, 2020.

[11] James O Berger, Brunero Liseo, and Robert L Wolpert. Integrated likelihood methods for eliminating nuisance parameters. *Statistical Science*, 14(1):1–28, 1999.

[12] Mario Bertero, Patrizia Boccacci, and Christine De Mol. *Introduction to inverse problems in imaging*. CRC press, 2021.

[13] Christopher Bishop, Markus Svensen, and Christopher Williams. Em optimization of latent-variable density models. *Advances in neural information processing systems*, 8, 1995.

[14] Wei Cai, Xiaoguang Li, and Lizuo Liu. A phase shift deep neural network for high frequency approximation and wave problems. *SIAM Journal on Scientific Computing*, 42(5):A3285–A3312, 2020.

[15] Olivier Cappé and Eric Moulines. On-line expectation–maximization algorithm for latent data models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 71(3):593–613, 2009.

[16] Jianfei Chen, Jun Zhu, Yee Whye Teh, and Tong Zhang. Stochastic expectation maximization with variance reduction. *Advances in Neural Information Processing Systems*, 31, 2018.

[17] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

[18] Claire Donnat, Axel Levy, Frederic Poitevin, Ellen D Zhong, and Nina Miolane. Deep generative modeling for volume reconstruction in cryo-electron microscopy. *Journal of structural biology*, 214(4):107920, 2022.

[19] Jacques Dubochet, Joachim Frank, and Richard Henderson. The nobel prize in chemistry 2017. *Nobel Media AB*, 2017.

[20] Rick Durrett. *Probability: theory and examples*, volume 49. Cambridge university press, 2019.

[21] Murat A Erdogdu and Andrea Montanari. Convergence rates of sub-sampled newton methods. *Advances in Neural Information Processing Systems*, 28, 2015.

[22] Zhou Fan, Yi Sun, Tianhao Wang, and Yihong Wu. Likelihood landscape and maximum likelihood estimation for the discrete orbit recovery model. *Communications on Pure and Applied Mathematics*, 76(6):1208–1302, 2023.

[23] Hassan Foroosh, Josiane B Zerubia, and Marc Berthod. Extension of phase correlation to subpixel registration. *IEEE transactions on image processing*, 11(3):188–200, 2002.

[24] Joachim Frank. *Three-dimensional electron microscopy of macromolecular assemblies: visualization of biological molecules in their native state*. Oxford University Press, 2006.

[25] Adrien Gallet, Samuel Rigby, TN Tallman, Xiangxiong Kong, Iman Hajirasouliha, Andrew Liew, Dong Liu, Liang Chen, Andreas Hauptmann, and Danny Smyl. Structural engineering from an inverse problems perspective. *Proceedings of the Royal Society A*, 478(2257):20210526, 2022.

[26] Roberto Gil-Pita, Manuel Rosa-Zurera, P Jarabo-Amores, and Francisco López-Ferreras. Using multilayer perceptrons to align high range resolution radar signals. In *International Conference on Artificial Neural Networks*, pages 911–916. Springer, 2005.

[27] Timothy Grant, Alexis Rohou, and Nikolaus Grigorieff. cis tem, user-friendly software for single-particle image processing. *elife*, 7:e35383, 2018.

[28] Yael Harpaz and Yoel Shkolnisky. Three-dimensional alignment of density maps in cryo-electron microscopy. *Biological Imaging*, 3:e8, 2023.

[29] Ayelet Heimowitz, Nir Sharon, and Amit Singer. Centering noisy images with application to cryo-EM. *SIAM Journal on Imaging Sciences*, 14(2):689–716, 2021.

[30] A Jiménez-Moreno, D Střelák, J Filipovič, JM Carazo, and CÓS Sorzano. Deepalign, a 3D alignment method based on regionalized deep learning for cryo-EM. *Journal of Structural Biology*, 213(2):107712, 2021.

[31] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in neural information processing systems*, 26, 2013.

[32] Yuehaw Khoo, Sounak Paul, and Nir Sharon. Deep neural-network prior for orbit recovery from method of moments. *Journal of Computational and Applied Mathematics*, 444:115782, 2024.

[33] Dari Kimanius, Liyi Dong, Grigory Sharov, Takanori Nakane, and Sjors HW Scheres. New tools for automated cryo-EM single-particle analysis in RELION-4.0. *Biochemical Journal*, 478(24):4169–4185, 2021.

[34] Ritesh Kolte, Murat Erdogdu, and Ayfer Ozgur. Accelerating svrg via second-order information. In *NIPS workshop on optimization for machine learning*, 2015.

[35] Xiangyin Kong, Xiaoyu Jiang, Bingxin Zhang, Jinsong Yuan, and Zhiqiang Ge. Latent variable models in the era of industrial big data: Extension and beyond. *Annual Reviews in Control*, 54:167–199, 2022.

[36] Werner Kühlbrandt. The resolution revolution. *Science*, 343(6178):1443–1444, 2014.

[37] Serge Lang. *Complex analysis*, volume 103. Springer Science & Business Media, 2013.

[38] Kenneth Lange. A quasi-newton acceleration of the em algorithm. *Statistica sinica*, pages 1–18, 1995.

[39] Daniel Lesnic. *Inverse problems with applications in science and engineering*. Chapman and Hall/CRC, 2021.

[40] Axel Levy, Frédéric Poitevin, Julien Martel, Youssef Nashed, Ariana Peck, Nina Miolane, Daniel Ratner, Mike Dunne, and Gordon Wetzstein. CryoAI: Amortized inference of poses for ab initio reconstruction of 3D molecular volumes from real cryo-EM images. *arXiv preprint arXiv:2203.08138*, 2022.

[41] Vladan Lučić, Alexander Rigort, and Wolfgang Baumeister. Cryo-electron tomography: the challenge of doing structural biology in situ. *Journal of Cell Biology*, 202(3):407–419, 2013.

[42] Geoffrey J McLachlan and Thriyambakam Krishnan. *The EM algorithm and extensions*. John Wiley & Sons, 2007.

[43] Ankur Moitra and Alexander S Wein. Spectral methods from tensor networks. In *Proceedings of the 51st Annual ACM Symposium on Theory of Computing*, pages 926–937. ACM, 2019.

[44] Takanori Nakane, Dari Kimanius, Erik Lindahl, and Sjors HW Scheres. Characterisa-

tion of molecular motions in cryo-EM single-particle data by multi-body refinement in RELION. *Elife*, 7:e36861, 2018.

[45] Takanori Nakane, Abhay Kotecha, Andrija Sente, Greg McMullan, Simonas Masiulis, Patricia MGE Brown, Ioana T Grigoras, Lina Malinauskaite, Tomas Malinauskas, Jonas Miehling, et al. Single-particle cryo-em at atomic resolution. *Nature*, 587(7832):152–156, 2020.

[46] F. Natterer. *The Mathematics of Computerized Tomography*. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 2001.

[47] RG Newton. Inverse problems in physics. *SIAM Review*, 12(3):346–356, 1970.

[48] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, New York, NY, USA, 2e edition, 2006.

[49] Evgenii Aleksandrovich Obornev, Ivan Evgen'evich Obornev, Evgenij Anatoljevich Rodionov, and Mikhail Il'ich Shimelevich. Application of neural networks in nonlinear inverse problems of geophysics. *Computational Mathematics and Mathematical Physics*, 60:1025–1036, 2020.

[50] Abbas Ourmazd. Cryo-em, xfels and the structure conundrum in structural biology. *Nature methods*, 16(10):941–944, 2019.

[51] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[52] Amelia Perry, Jonathan Weed, Afonso S Bandeira, Philippe Rigollet, and Amit Singer. The sample complexity of multireference alignment. *SIAM Journal on Mathematics of Data Science*, 1(3):497–517, 2019.

[53] Eric F Pettersen, Thomas D Goddard, Conrad C Huang, Gregory S Couch, Daniel M Greenblatt, Elaine C Meng, and Thomas E Ferrin. Ucsf chimera—a visualization system for exploratory research and analysis. *Journal of computational chemistry*, 25(13):1605–1612, 2004.

[54] Boris T Polyak. Newton's method and its use in optimization. *European Journal of Operational Research*, 181(3):1086–1096, 2007.

[55] Ali Punjani and David Fleet. Advances in modelling continuous heterogeneity from single particle cryo-EM data. *Foundations of Crystallography*, 77:A235–A235, 2021.

[56] Ali Punjani, John L Rubinstein, David J Fleet, and Marcus A Brubaker. cryoSPARC: algorithms for rapid unsupervised cryo-EM structure determination. *Nature methods*, 14(3):290, 2017.

[57] Dirk Robinson, Sina Farsiu, and Peyman Milanfar. Optimal registration of aliased images using variable projection with applications to super-resolution. *The Computer Journal*, 52(1):31–42, 2009.

[58] Vladimir Gavrilovich Romanov. *Inverse problems of mathematical physics*. Walter de Gruyter GmbH & Co KG, 2018.

[59] David M Rosen, Luca Carlone, Afonso S Bandeira, and John J Leonard. A certifiably correct algorithm for synchronization over the special euclidean group. In *Algorithmic Foundations of Robotics XII: Proceedings of the Twelfth Workshop on the Algorithmic Foundations of Robotics*, pages 64–79. Springer, 2020.

[60] Eitan Rosen and Yoel Shkolnisky. Common lines ab-initio reconstruction of D_2-symmetric molecules. *SIAM Journal on Imaging Sciences*, 2020. To appear.

[61] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[62] Malcolm Sambridge and Klaus Mosegaard. Monte carlo methods in geophysical inverse problems. *Reviews of Geophysics*, 40(3):3–1, 2002.

[63] Sjors HW Scheres. Processing of structurally heterogeneous cryo-EM data in RELION. In *Methods in enzymology*, volume 579, pages 125–157. Elsevier, 2016.

[64] Sjors HW Scheres, Mikel Valle, Rafael Nuñez, Carlos OS Sorzano, Roberto Marabini, Gabor T Herman, and Jose-Maria Carazo. Maximum-likelihood multi-reference refinement for electron microscopy images. *Journal of molecular biology*, 348(1):139–149, 2005.

[65] Fanhua Shang, Kaiwen Zhou, Hongying Liu, James Cheng, Ivor W Tsang, Lijun Zhang, Dacheng Tao, and Licheng Jiao. Vr-sgd: A simple stochastic variance reduction method for machine learning. *IEEE Transactions on Knowledge and Data Engineering*, 32(1):188–202, 2018.

[66] Nir Sharon, Joe Kileel, Yuehaw Khoo, Boris Landa, and Amit Singer. Method of moments for 3D single particle ab initio modeling with non-uniform distribution of viewing angles. *Inverse Problems*, 36(4):044003, 2020.

[67] Sviatlana Shashkova and Mark C Leake. Single-molecule fluorescence microscopy review: shedding new light on old problems. *Bioscience reports*, 37(4):BSR20170031, 2017.

[68] Yoel Shkolnisky and Amit Singer. Viewing direction estimation in cryo-EM using synchronization. *SIAM journal on imaging sciences*, 5(3):1088–1110, 2012.

[69] Amit Singer. Mathematics for cryo-electron microscopy. In *Proceedings of the International Congress of Mathematicians*, volume 4, pages 4013–4032, 2018.

[70] Amit Singer and Yoel Shkolnisky. Three-dimensional structure determination from common lines in cryo-EM by eigenvectors and semidefinite programming. *SIAM journal on imaging sciences*, 4(2):543–572, 2011.

[71] Devika Sirohi, Zhenguo Chen, Lei Sun, Thomas Klose, Theodore C Pierson, Michael G Rossmann, and Richard J Kuhn. The 3.8 Å resolution cryo-EM structure of Zika virus. *Science*, 352(6284):467–470, 2016.

[72] Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. Solving inverse problems in medical imaging with score-based generative models. *arXiv preprint arXiv:2111.08005*, 2021.

[73] Douglas L Theobald and Phillip A Steindel. Optimal simultaneous superpositioning of multiple structures with missing data. *Bioinformatics*, 28(15):1972–1979, 2012.

[74] Alexander S Wein. *Statistical estimation in the presence of group actions*. PhD thesis, Massachusetts Institute of Technology, 2018.

[75] Robert S Womersley. Efficient spherical designs with good geometric properties. *Contemporary computational mathematics-A celebration of the 80th birthday of Ian Sloan*, pages 1243–1285, 2018.

[76] Shenping Wu, Jean-Paul Armache, and Yifan Cheng. Single-particle cryo-em data acquisition by using direct electron detection camera. *Journal of Electron Microscopy*, 65(1):35–41, 2015.

[77] Andy Zhang, Oscar Mickelin, Joe Kileel, Eric J Verbeke, Nicholas F Marshall, Marc Aurèle Gilles, and Amit Singer. Moment-based metrics for molecules computable from cryo-em images. *Biological Imaging*, pages 1–22, 2024.

[78] Ellen D Zhong, Tristan Bepler, Bonnie Berger, and Joseph H Davis. CryoDRGN: reconstruction of heterogeneous cryo-EM structures using neural networks. *Nature methods*, 18(2):176–185, 2021.

[79] Ellen D Zhong, Adam Lerer, Joseph H Davis, and Bonnie Berger. CryoDRGN2: Ab initio neural reconstruction of 3D protein structures from real cryo-EM images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4066–4075, 2021.

[80] Jasenko Zivanov, Takanori Nakane, Björn O Forsberg, Dari Kimanius, Wim JH Hagen, Erik Lindahl, and Sjors HW Scheres. New tools for automated high-resolution cryo-em structure determination in RELION-3. *Elife*, 7:e42166, 2018.

[81] Joris Portegies Zwart, René van der Heiden, Sjoerd Gelsema, and Frans Groen. Fast translation invariant classification of HRR range profiles in a zero phase representation. *IEE Proceedings-Radar, Sonar and Navigation*, 150(6):411–418, 2003.

APPENDIX A

ARCHITECTURE OF NEURAL NETWORKS

In this appendix, we describe the details of the NN for both MRA and cryo-EM in Chapters 2 and 3. To facilitate the discussion, we first define $\texttt{conv1D}_{w,c}$ to be a 1D convolutional layer with periodic padding, kernel window size w and channel number c. In a similar way, we also denote a 2D convolutional layer with window size $w \times w$ and channel number c as $\texttt{conv2D}_{w,c}$. Furthermore we define $\texttt{input1D}_{\ell,c}$ to be an input layer that prepares the input as a length ℓ 1D vector field with channel number c. We then define a fully connected layer $\texttt{full}_w$ that takes an input vector field and output a vector with size w. The nonlinearities we use in this chapter are leaky ReLu (LReLu) nonlinear activation with parameter 0.02, $\tanh(\cdot)$ function, and just linear activation (without nonlinearities). We make no distinction between real or complex input, since changing real to complex input only requires doubling the input or output channel number.

A.1 Multireference alignment

In the MRA case, we present the proposed architecture for the encoder ξ_θ. An illustration of ξ_θ^ρ is presented in Figure (A.1), and the same architecture is used for ξ_θ^v. The input layers $\texttt{input1D}_{n,1}$ and $\texttt{input1D}_{n,n}$ take the moments as inputs. After a few layers of `conv1D`, we stack the output of the upper branch and lower branch in Figure (A.1) together into a 1D vector field of length n and 6 channels. Then after a few more layers of CNN `conv1D` and fully connected layers `full`, we output z_ρ.

A.2 Cryo-EM

For ξ_θ in Chapter 2, the constituent encoders ξ_θ^ρ and ξ_θ^v are very similar to the one presented in Figure (A.1) for MRA, except we replace all `conv1D` with `conv2D` with the same window

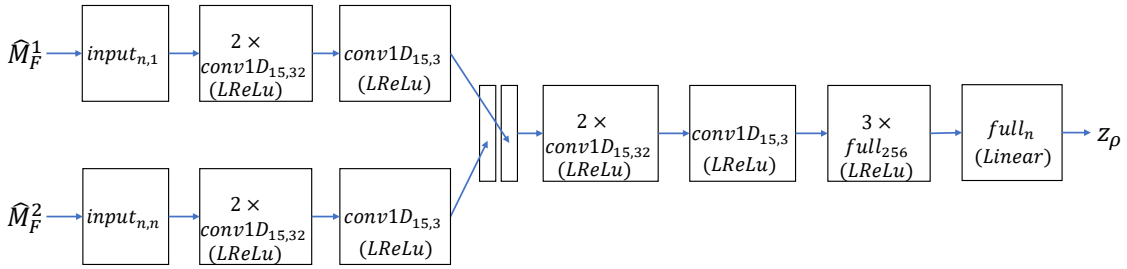

Figure A.1: Architecture of ξ_θ^ρ in case of MRA

sizes and channel numbers. In Chapter 3, a further change in the encoder ξ_θ^ρ is that it outputs $(z_\rho, \hat{\eta})$ instead of simply z_ρ.

As for $\widehat{v}_\phi$ of Chapter 2 and $\widehat{v}_{\text{Fnet}}$ of Chapter 3, currently, they are chosen to be the FourierNet of [40]. FourierNet finds success in representing the Fourier transforms of three-dimensional volumes of molecules and other volumes arising in nature, with values that often span multiple orders of magnitude. The main point of such a representation is that, instead of approximating $v(x)$ directly by an NN, it is often easier to approximate its Fourier coefficients $\widehat{v}(x)$ by an NN on k-space when $v(x)$ exhibits oscillatory patterns. This is also similar to the approach taken in [14] for solving high-frequency wave equations. More precisely, it lets

$$\widehat{v}(k) \approx \widehat{v}_\phi(k) = a_{\phi_1}(k) \exp(i b_{\phi_2}(k)) \tag{A.1}$$

with two NNs $a_{\phi_1}(k) \in \mathbb{C}$ and $b_{\phi_2}(k) \in \mathbb{C}$ where a_{ϕ_1} gives the amplitude of the Fourier coefficients and b_{ϕ_2} gives the phase variations. By representing v in Fourier domain instead of real domain, one can bypass the oscillatory pattern caused by the Fourier series $\exp(ikx)$ in $v(x) = \sum_k \widehat{v}(k) \exp(ikx)$. More details regarding the architecture, its effectiveness, and its memory requirements are provided in [40].