INAUGURAL – DISSERTATION

zur

Erlangung der Doktorwürde

der

Gesamtfakultät für Mathematik, Ingenieur- und Naturwissenschaften

der

Ruprecht–Karls–Universität

Heidelberg

vorgelegt von
M.Sc. Bastian Benjamin Boll
aus Leimen

Tag der mündlichen Prüfung:

On Structured Prediction of Discrete Data: Geometry and Statistical Learning

Advisor: Prof. Dr. Christoph Schnörr

Abstract

Structured prediction is the task of jointly predicting realizations of multiple coupled random variables. This statistical problem is central to many advanced applications of deep learning, including image segmentation and graph node classification. This thesis presents a two-pronged study of predicting structured discrete data, exploring geometric aspects and statistical learning. On the geometric side, we first interpret distributions of independent discrete random variables as points on a product manifold of probability simplices. We find that this manifold is isometrically embedded into the meta-simplex of joint probability distributions. This finding illuminates the relationship between inference dynamics on the product manifold, called assignment flows, and replicator dynamics on the meta-simplex. The former can be seen as the replicator dynamics of multi-population games and the constructed embedding formally reduces them to high-dimensional single-population game dynamics. Based on these geometric insights, we propose two types of generative models for discrete data by facilitating measure transport through randomized assignment flows. The first approximates a given energy-based model, while the second is learned directly from data. Experiments on image segmentation data illustrate the viability of the proposed method. With regard to statistical learning, we explore current methods in PAC-Bayesian risk certification and propose a classification approach with favorable computational properties. Further, we develop a novel PAC-Bayesian risk bound for structured prediction, which can account for generalization even from a single structured datum. The lack of independent data is addressed by distilling the coupling structure of the joint data distribution, given as a Knothe-Rosenblatt rearrangement of a reference measure, allowing for the use of modern concentration of measure results.

Zusammenfassung

Strukturierte Vorhersage bezeichnet das Problem, Realisierungen mehrerer gekoppelter Zufallsvariablen vorherzusagen. Dieses statistische Problem ist von zentraler Bedeutung für eine Vielzahl komplexer Anwendungen des tiefen Lernens, einschließlich der Bildsegmentierung und Klassifizierung von Graphknoten. Diese Arbeit beleuchtet die strukturierte Vorhersage diskreter Daten sowohl aus geometrischer Perspektive als auch in Bezug auf statistisches Lernen. Auf der geometrischen Seite interpretieren wir zunächst Verteilungen unabhängiger diskreter Zufallsvariablen als Punkte einer Produktmannigfaltigkeit von Simplexen. Wir stellen fest, dass diese Mannigfaltigkeit isometrisch in das Meta-Simplex multivariater Wahrscheinlichkeitsverteilungen eingebettet ist. Diese Erkenntnis beleuchtet die Beziehung zwischen Inferenzdynamiken auf der Produktmannigfaltigkeit, sogenannter Zuweisungsflüsse, und Replikator Dynamiken im Meta-Simplex. Erstere können als Replikator Dynamiken mehrerer Populationen betrachtet werden, wobei die konstruierte Einbettung diese formal auf Spieldynamiken einer einzelnen, hochdimensionalen Population reduziert. Basierend auf diesen geometrischen Einsichten entwickeln wir zwei generative Modelle für diskrete Daten, die Maßtransport durch Randomisierung von Zuweisungsflüssen realisieren. Das erste Modell approximiert ein gegebenes Energiemodell, während das zweite direkt aus Daten gelernt wird. Experimente mit Bildsegmentierungsdaten veranschaulichen die Anwendbarkeit der vorgeschlagenen Methoden. In Bezug auf statistisches Lernen explorieren wir aktuelle PAC-Bayessche Methoden und stellen einen Ansatz für Klassifikationsprobleme vor, der günstige Berechnung erlaubt. Darüber hinaus entwickeln wir eine PAC-Bayessche Schranke an die Kosten strukturierter Prädiktoren, welche Generalisierung sogar aus einem einzelnen strukturierten Datum beschreiben kann. Hierbei wird die Abwesenheit von statistisch unabhängigen Daten durch explizite Extraktion von deren Kopplungsstruktur berücksichtigt. Der Konstruktion liegt die Annahme einer Datenverteilung zugrunde, die durch Knothe-Rosenblatt Umordnung eines Referenzmaßes gegeben ist, was moderne Ergebnisse zum Phänomen der Konzentration des Maßes zugänglich macht.

Acknowledgements

Many people have contributed to this thesis in different ways. First, I want to thank my advisor, Prof. Christoph Schnörr, for his continuous guidance throughout my studies. He has always been generous with his time and resources and quick in responding to inquiries. I have greatly benefitted from his mathematical insight, references to literature, and his help navigating the academic landscape. He has cultivated a great work environment where I always felt like new ideas were valued.

I also thank many current and former colleagues at the Image and Pattern Analysis group and the Mathematical Imaging Group in Heidelberg. Starting out, I was warmly welcomed by Artjom Zern, Matthias Zisler, Alexander Zeilmann, Barbara Werner, Dmitrij Sitenko, Fabrizio Savarino, Jan Plier, Lukas Kiefer and Ruben Hühnerbein. Every one of them has shared insightful perspectives, explained new concepts, and helped me find a place in research—all while being simply great company, which I continue to treasure. Likewise, I want to thank all colleagues I have had the pleasure of meeting since then, including Jonathan Schwarz, Daniel Gonzalez-Alvarado, Jonas Cassel, Yara Elshiaty, Christian Homeyer, Felix Draxler, and Sebastian Müller. Over the course of projects, lunches, coffee breaks, dinners, conferences, tennis matches, and more, I learned a lot from you and had a great time.

I want to thank Evelyn Wilhelm, Jiannis Giatagantzidis, and Jutta Wiech for their help with administrative matters. I warmly thank Barbara Werner not only for administrative help but also for initiating events, sharing coffee, and having a good time. She was a highly appreciated part of our team, and we all miss her.

Outside academia, I thank my close friends and family, especially my parents, Arne Boll and Sonja Weinreich-Boll, for much love and support. My deepest appreciation goes to Alena Fischer. I am lucky to have her as a teammate in life, and throughout the years spent preparing this thesis, she always had my back.

This work is funded by the Deutsche Forschungsgemeinschaft (DFG), grant SCHN 457/17-1, within the priority programme SPP 2298: “Theoretical Foundations of Deep Learning”. This work is funded by the Deutsche Forschungsgemeinschaft (DFG) under Germany’s Excellence Strategy EXC-2181/1 - 390900948 (the Heidelberg STRUCTURES Excellence Cluster).

Contents

List of Publications

The following conference papers and preprints are directly related to the content of this thesis.

1. Boll, B., & Schnörr, C. (2023). On Certified Generalization in Structured Prediction. Advances in Neural Information Processing Systems, 36, 30499–30517. Curran Associates.

2. Boll, B., Zeilmann, A., Petra, S., & Schnörr, C. (2023). Self-Certifying Classification by Linearized Deep Assignment. Proceedings in Applied Mathematics and Mechanics, 23(1), e202200169. doi:10.1002/pamm.202200169

3. Boll, B., Schwarz, J., & Schnörr, C. (2021). On the Correspondence between Replicator Dynamics and Assignment Flows. 8th International Conference on Scale Space and Variational Methods in Computer Vision, 12679, 373–384. Springer.

4. Boll, B., Schwarz, J., Gonzalez-Alvarado, D., Sitenko, D., Petra, S., & Schnörr, C. (2023). Modeling Large-scale Joint Distributions and Inference by Randomized Assignment. 9th International Conference on Scale Space and Variational Methods in Computer Vision, 14009, 730–742. Springer.

5. Boll, B., Cassel, J., Albers, P., Petra, S., & Schnörr, C. (2024). A Geometric Embedding Approach to Multiple Games and Multiple Populations. Preprint arXiv:2401.05918 (submitted).

6. Boll, B., Gonzalez-Alvarado, D., & Schnörr, C. (2024). Generative Modeling of Discrete Joint Distributions by E-Geodesic Flow Matching on Assignment Manifolds. Preprint arXiv:2402.07846.

An elaboration of the paper 6 is in preparation for journal submission.

The author has also made significant contributions to the following further publications over the course of preparing this thesis.

7. Schwarz, J., Cassel, J., Boll, B., Gärttner, M., Albers, P., & Schnörr, C. (2023). Quantum State Assignment Flows. Entropy, 25(9), 1253.

8. Schwarz, J., Boll, B., Sitenko, D., Gonzalez-Alvarado, D., Gärttner, M., Albers, P., & Schnörr, C. (2023). Quantum State Assignment Flows. 9th International Conference on Scale Space and Variational Methods in Computer Vision, 14009, 730–742. Springer.

9. Sitenko, D., Boll, B., & Schnörr, C. (2023). A Nonlocal Graph-PDE and Higher-Order Geometric Integration for Image Labeling. SIAM Journal on Imaging Sciences, 16(1), 501–567.

10. Sitenko, D., Boll, B., & Schnörr, C. (2022). Assignment Flows and Nonlocal PDEs on Graphs. In Bauckhage, C., Gall, J. & Schwing, A. (Eds.), Pattern Recognition. DAGM GCPR 2021. Lecture Notes in Computer Science, vol 13024, 498–512. Springer.

11. Sitenko, D., Boll, B., & Schnörr, C. (2021). Assignment Flow For Order-Constrained OCT Segmentation. International Journal of Computer Vision, 129, 3088–3118. Springer.

12. Sitenko, D., Boll, B., & Schnörr, C. (2021). Assignment Flow for Order-Constrained OCT Segmentation. In Z. Akata, A. Geiger, & T. Sattler (Eds.), Pattern Recognition. DAGM GCPR 2020. Lecture Notes in Computer Science, vol 12544, 58–71. Springer.

1 Introduction

The 21st century has witnessed a new rise of deep learning, evolving from a specialized topic within statistics to a pervasive subject with applications in diverse fields. This trend has accelerated in recent years, with advanced generative models for images and natural language reaching beyond academia to gain broad mainstream appeal and awareness among the general public. The widespread adoption and evident utility of tools like ChatGPT and DALL-E – commercial products built around large language models and latent diffusion models – have also prompted concerns about their societal implications. Issues such as environmental impacts, the exacerbation of wealth disparities, discrimination through social biases embedded in data, and fear of job displacement have become increasingly pressing as these technologies permeate more aspects of daily life.

At the heart of this technological revolution lie impressive feats of engineering, accompanied by a large body of theoretical work. For example, a core challenge in training large models involves reliably predicting effective hyperparameters and the subsequent performance of large models based on smaller ones [1]. A way to achieve this was proposed by [217], as a first application emerging from the *tensor programs* series of theoretical works [220, 218, 219, 221, 217, 222]. Despite these achievements, fundamental research questions remain elusive. For instance, the *double descent* phenomenon [151], a counterintuitive increase in performance for large models, is at the very core of deep learning and only beginning to be illuminated in simple scenarios. Breakthroughs in understanding the mathematical foundations of deep learning could be invaluable, not only for advancing technical capabilities but also to inform public discourse on the impacts of emerging technologies and potentially to help steer developments toward more positive outcomes.

A core challenge at the theoretical foundation of deep learning concerns the development of statistical learning theories, which can reliably predict the performance of overparameterized models and guide their training in a principled way. PAC-Bayesian theory [189, 143, 142] has gained considerable momentum in this field, partly due to the demonstration of non-vacuous risk certificates for deep classifiers [66]. However, models dealing with complex modalities such as natural language, graphs, and images are often at odds with the assumption of independently drawn data – a cornerstone of the most potent statistical theories.

These domains fall into the category of *structured prediction* problems, where data are drawn from the joint distribution of multiple *coupled* random variables. Take image

segmentation as an example, where labels for all pixels are naturally predicted jointly rather than separately. If all neighbors of any given pixel are already classified as belonging to one class, it is likely that the pixel in question also belongs to this class. Modeling image pixels as nodes of a grid graph, this property of neighbors tending to share the same class is called *homophily*, a prevalent property of many real-world graph datasets. For instance, consider citation graphs such as those in [26], where each node represents an academic publication, and two nodes are connected if one work is cited in the other. If publications are classified into fields of study, the citation graph has high homophily. This is because authors are likely to cite works predominantly in their own field of study. Conversely, graphs in which neighbors are likely to have different classes are called *heterophilic*. Citation graphs can also be heterophilic; for example, if the class to be predicted is the year of publication, the underlying pattern being that most cited works are not concurrent. Both homophily and heterophily in graphs are examples of interdependencies between nodes. Thus, formalizing the label on each node as a random variable, their joint distribution induces complex, possibly global dependencies.

Similar long-range dependencies are also present in natural language corpora. It is easy to see that tokens within a sentence or paragraph are coupled. For instance, anaphora, expressions that refer back to phrases used elsewhere in the text, appear frequently in natural language [163]. Long-range dependencies within a large corpus are more subtle. Still, it can be argued that tokens can exhibit dependency at arbitrary distances, even if the corpus consists of multiple, supposedly independent documents, possibly in multiple different languages. This interconnectedness arises because natural language is a product of human culture. For example, a single influential work of literature may not only be quoted but become part of the perspective of subsequent authors, influencing their writing in subtle ways. For this reason, working with natural language entails a fine-grained analysis of statistical dependencies between tokens – the hallmark of structured prediction problems.

Complex applications like the modeling of natural language serve as motivation for an in-depth study of structured prediction. However, in this thesis, we do not focus on application-specific aspects. Our primary objects of interest are joint distributions of n coupled random variables, with each variable taking values in the discrete set $[c] = \{1, \dots, c\}$. Although somewhat abstract, this formalization captures many aspects of complex real-world data distributions without introducing a significant notational burden.

1.1 Contribution and Organization

In the first part of this thesis, we develop a geometric formalization and intuition for structured prediction, focusing in particular on discrete random variables. The distribution of a discrete variable is interpreted as a point on a geometric domain called *probability simplex*. We discuss several related concepts and notations, emerging primarily from the field of *information geometry*, in the preliminary section 2.1. The second preliminary section 2.2 summarizes elements of statistical learning theory, which become relevant in the second part of this thesis.

Chapter 3 introduces *assignment flows*, dynamical systems on the product manifold of multiple probability simplices with a natural shape relative to the underlying information geometry. We present a first-principles approach to constructing these systems, contextualizing them relative to probabilistic graphical models and replicator dynamics in game theory. Assignment flows are a central concept in this thesis and have an intricate link to structured prediction. Initially, they evolve a state that represents the joint distribution of *independent* random variables due to the product structure of the underlying domain. However, assignment flows also facilitate interaction between different components over time, which implicitly encodes coupling.

Chapter 4 studies the relationship between factorizing joint distributions and general joint distributions of possibly coupled random variables. We show that factorizing distributions form a submanifold, which is isometrically embedded. Building on this, the main result of Chapter 4 is that, under the constructed embedding of factorizing distributions, assignment flows transform to replicator dynamics on the meta-simplex of general joint distributions. We further explicitly compute the shape of interaction, called the payoff function, driving these replicator dynamics. The results of this chapter clarify the relationship between assignment flows and established concepts in game theory by painting them as replicator dynamics on multiple populations. From this perspective, the proposed embedding constitutes a formal reduction of multi-population to single-population replicator dynamics. We demonstrate the use of this formal reduction, in conjunction with our newly developed formalism, by transferring established results on the asymptotic behavior of replicator dynamics from the single-population to the multi-population setting.

Chapter 5 builds on the geometric foundation developed in the previous chapter to parameterize large classes of joint distributions through assignment flows. The proposed method leverages randomization to explain how the interaction between components in the product distribution domain of assignment flows can generate *coupling* between random variables. To this end, we propose two distinct flavors of randomization and two methods of learning joint distributions. The first, described in Section 5.2, is a way to approximate distributions with a known shape as an energy-based model. A core technical aspect of the construction concerns differentiable approximation of model entropy. We employ a class of assignment flows with provable convergence to discrete Dirac measures and, leveraging this asymptotic behavior, we propose an estimator of model entropy that becomes exact in the limit of long integration time. In Section 5.3, we use a variant of randomized assignment flows to learn a generative model directly from data. The resulting model can be regarded as a continuous normalizing flow and aligns naturally with *dequantization*, an approach to modeling discrete data commonly used in conjunction with normalizing flows. A further similarity to established normalizing flows is the parameterization of payoff functions driving assignment dynamics, which we choose as a deep neural network. The network parameters are learned such that assignment dynamics perform measure transport on the underlying product manifold by applying a recently proposed Riemannian generalization of *flow matching*. In contrast to the traditional method of maximizing data likelihood, which minimizes relative entropy to the target distribution, flow matching

is a simulation-free alternative with improved computational efficiency. In the setting of discrete data, where likelihood under the model can no longer be computed directly by an instantaneous change of variables, we develop an importance sampling approach for computing likelihood. Experiments on image segmentation datasets demonstrate the viability of the proposed generative model for discrete data.

Turning to the statistical part of this thesis, we propose a self-certified method for image classification in Chapter 6. Unlike traditional methods, which evaluate the out-of-sample performance on held-out test data, self-certified approaches use all available data to simultaneously learn classifiers and certify their risk on unseen data from the same distribution. This is based on the PAC-Bayesian risk certification paradigm, which can achieve non-vacuous high probability upper bounds on the risk of deep stochastic classifiers. Within this framework, we construct a hypothesis class of stochastic classifiers based on linearizing assignment flows. The proposed architecture keeps most parameters deterministic and employs a data-dependent affine linear transformation of stochastic parameters. By this construction, classification logits always follow a normal distribution, and we further propose an efficient numerical method to compute respective moments. Experiments on two image classification datasets illustrate that the proposed approach is computationally efficient and on par with current PAC-Bayesian methods in terms of certificate tightness.

In Chapter 7, we explore the PAC-Bayesian approach further, to develop self-certified methods for structured prediction. Unlike the simpler case of classification, independence of data cannot be assumed in structured prediction. We argue that this justifies the need to make assumptions about the data distribution. Consequently, we employ *Knothe-Rosenblatt rearrangement*, a particular form of measure transport that exists uniquely for a wide range of distributions. Our approach leverages the triangular structure of the Knothe-Rosenblatt transport map to distill the coupling of random variables present in the joint distribution of data into a *Wasserstein dependency matrix*. Using this matrix, we apply modern concentration of measure theory, leading to a moment-generating function bound. This, in turn, serves as the foundation for constructing a novel PAC-Bayesian bound for structured prediction that accounts for generalization from a single example.

Chapter 8 concludes the thesis and points out possible directions for future work.

1.2 Background and Related Work

1.2.1 Assignment Flows

This thesis explores methods within the assignment flows paradigm. The underlying perspective is to regard discrete probability distributions as points on a probability simplex, conceptualized as a statistical manifold by information geometry [7, 6, 12]. Building on these notions, [11] introduced assignment flows as dynamical systems on the product manifold of multiple probability simplices designed to tackle image labeling problems. Subsequent research has studied fundamental properties of assignment flows, including

effective numerical integration [225] and asymptotic behavior [227].

In contrast to probabilistic graphical models, assignment flows are smooth processes, which notably simplifies their integration with deep neural networks. The first step in this direction was made by [98], who have developed an adjoint integration method that efficiently learns the parameters driving assignment flows from data. The method can be categorized as a neural ordinary differential equation [43], albeit carrying a specialized structure that can help integrate prior knowledge about a problem domain. For example, [190] devise a segmentation technique for volumetric data acquired via optical coherence tomography. The approach is a two-stage architecture, combining a deep neural network for extracting semantic features with an assignment flow that enforces a layer ordering constraint relevant to the medical application being considered.

In this thesis, we use assignment flows to facilitate measure transport. The closest related work in this regard is the uncertainty quantification method proposed by [76], which computes the pushforward of an initial, uncertain state under linearized assignment flow dynamics. For a comprehensive overview of assignment flows, we refer to [185].

1.2.2 Generative Modelling

The most successful generative models of discrete sequence data [35, 54] are autoregressive, predominantly employing transformer architectures [211]. The idea of autoregressive modelling is to divide the complexity of joint data distributions by dividing into a sequence of next-token probability distributions, conditioned on previous context. This is well-adapted to the structure of sequence data, including text. However, it is less natural to model discrete image data autoregressively, because no natural ordering of tokens exists a priori and choosing an arbitrary order leads to complex, global dependencies in a long sequence of pixels, obfuscating the original two-dimensionality of the data in the process. As a result, it is difficult to scale such an approach to larger images, considering the quadratic memory and compute complexity of transformers in the context length. Very recently, selective state space models [79] were proposed, adressing this scaling problem of transformers, but they are still autoregressive models designed for sequence data.

For the image modalities which we focus on in this thesis, an alternative to autoregressive modelling is modelling the generator of a dynamical system which gradually produces samples. Intuitively, this distributes the complexity of the joint distribution over an artificially created time axis, by breaking down the sampling process into multiple iterative steps. If steps are taken deterministically, the respective class of models is called normalizing flows [109, 157]. In contrast, stochastic steps are taken by diffusion models [223]. In both cases, time can be modelled as continuous [194, 43] or discrete [193, 60, 91]. Traditionally, normalizing flows are trained by maximizing the likelihood of data, thereby minimizing relative entropy to the data distribution. In contrast, a popular way to train diffusion models is score matching [195].

Recently, the line between normalizing flows and diffusion models has been blurred. Partly, this is due to the introduction of the probability flow ODE [194], deterministic dynamics which produce marginal probabilities matching those of the stochastic diffusion process. In addition to computational benefits [131], this allows to see diffusion as a

measure transport process equivalent to a continuous normalizing flow. Conversely, [125] introduce the *flow matching* approach for training normalizing flows. This is more efficient than traditional likelihood maximization, because it avoids costly simulation of sampling trajectories and their gradients. Additionally, [125] show that a special case of their construction leads to the probability flow ODE of [194].

In Section 5.3, we propose a generative model of discrete data built on the geometric principles underlying assignment flows. The proposed model is a normalizing flow on a Riemannian manifold trained by flow matching. To this best of our knowledge, this combination was only considered elsewhere in the concurrent work of [196]. However, both diffusion models and normalizing flows have previously been studied on Riemannian manifolds [97, 52, 130, 137, 18, 172] and both paradigms have also previously been applied to discrete data [205, 96, 44].

1.2.3 PAC-Bayesian Theory and Structured Prediction

PAC-Bayesian theory [189, 143, 142] has attracted significant interest in recent years, partly due to the demonstration of non-vacuous risk certificates for deep classifiers [66, 66, 123]. While several works have explored methods to relax the underlying assumptions of bounded loss [82, 81] and independent and identically distributed (i.i.d.) data [4], comparatively little attention has been directed toward structured prediction. An exception is the work [36], which offers a PAC-Bayesian perspective on the implicit loss embedding framework [46]. For an overview of PAC-Bayesian theory we refer to [38, 80, 3].

In Chapter 7, we continue a line of research started by [127, 128, 126], which aims to construct PAC-Bayesian risk bounds for structured prediction that account for generalization from a single structured datum. Instrumental to their analysis is the stability of inference and quantified dependence in the data distribution. The latter is expressed in terms of ϑ-mixing coefficients, the total variation distance between data distributions conditioned on fixed values for a subset of variables. For structured prediction with Hamming loss, a coupling of such conditional measures can be constructed [68] such that ϑ-mixing coefficients yield an upper-bound that allows to invoke concentration of measure arguments [112]. The result is a bound on a moment-generating function, which the authors employ in a subsequent PAC-Bayesian construction, achieving generalization from a single datum.

The underlying assumption of these previous works is that data are generated by a Markov random field (MRF). This model assumption is limiting because the assumed Markov properties likely do not hold for many real-world data distributions. In addition, it is challenging to work with MRFs computationally. Exact inference in MRFs is NP-hard [214], and thus learning, which often contains inference as a subproblem, presents significant computational roadblocks. Even once an MRF has been found that represents data reasonably well, numerical evaluation of the PAC-Bayesian risk certificate proposed in [126] will again be computationally complex.

1.3 Basic Notation

For vectors v, the exponential function $\exp(v)$ and logarithm $\log(v)$ apply componentwise. The symbol $\diamond$ is used to denote componentwise multiplication of vectors $v \diamond w$ and of matrices $A \diamond B$. Both vectors and matrices will also sometimes occur in fractions $\frac{v}{w}$, $\frac{A}{B}$, which denote componentwise division. Angled brackets $\langle \cdot, \cdot \rangle$ denote the standard inner product on Euclidean spaces and $\langle A, B \rangle = \operatorname{tr} A^\top B$ for matrices. Let $(\mathcal{X}, \Sigma)$ and $(\mathcal{X}', \Sigma')$ be measurable spaces and let μ, ν be measures on $\mathcal{X}$. Then $\mu \ll \nu$ denotes absolute continuity of μ with respect to ν. If $A \in \Sigma$, this means that $\nu(A) = 0$ implies $\mu(A) = 0$. In this case, the relative entropy of μ and ν

$$\mathrm{KL}(\mu, \nu) = \int_{\mathcal{X}} \mu(x) \log \frac{\mathrm{d}\mu}{\mathrm{d}\nu}(x) \mathrm{d}x \tag{1.1}$$

is well defined due to existence of the Radon-Nikodým derivative $\frac{\mathrm{d}\mu}{\mathrm{d}\nu}$, i.e. a density of μ relative to ν, which is unique up to sets of measure zero. For a measurable function $f \colon \mathcal{X} \to \mathcal{X}'$, the pushforward measure (on $\mathcal{X}'$) of μ under f is denoted by $f_\sharp \mu$. For $A' \in \Sigma'$, this is defined by $(f_\sharp \mu)(A') = \mu(f^{-1}(A'))$. For vectors v, w or matrices A, B of suitable dimension, we denote stacking into block matrices with square brackets

$$\begin{bmatrix} A & B \\ v^\top & w^\top \end{bmatrix}. \tag{1.2}$$

2 Preliminaries

Throughout this thesis, we assume the reader is familiar with established notation and commonly-used concepts of differential geometry, including Riemannian geometry. For general reference, we refer to [120, 103].

2.1 Probability Simplex

For any measure space $(\mathcal{X}, \Sigma, \lambda)$, let $\overline{\mathcal{P}}(\mathcal{X}, \Sigma, \lambda)$ denote the set of probability distributions with domain $\mathcal{X}$ and a density with respect to the base measure λ. In the following, we will use the shorthand $\overline{\mathcal{P}}(\mathcal{X}) = \overline{\mathcal{P}}(\mathcal{X}, \Sigma, \lambda)$ if the σ-algebra Σ and base measure λ are clear from context. In particular, if $\mathcal{X}$ is a Euclidean space, we take Σ as the Borel σ-algebra and λ as the Lebesgue measure. If $\mathcal{X}$ is a finite set, we take $\Sigma = 2^{\mathcal{X}}$ as the power set and λ as the counting measure. Further, we define

$$\mathcal{P}(\mathcal{X}) = \{p \in \overline{\mathcal{P}}(\mathcal{X}) \colon \lambda(A) > 0 \Rightarrow p(A) > 0 \; \forall A \in \Sigma\} \tag{2.1}$$

as the subset of distributions with *full support* on $\mathcal{X}$.

A large part of this thesis evolves around random variables which take values in a discrete set $[c] = \{1, \cdots, c\}$ and are governed by a probability distribution $p \in \overline{\mathcal{P}}([c])$. These discrete distributions can be written as probability vectors listing the c probabilities p_i of singletons $\{i\} \subseteq [c]$, $i \in [c]$ under p. The set of these probability vectors is called *probability simplex*

$$\Delta_c = \{p \in \mathbb{R}^c \colon p \geq 0, \quad \langle p, \mathbb{1}_c \rangle = 1\} \equiv \overline{\mathcal{P}}([c]). \tag{2.2}$$

If Δ_c is regarded as a subset of its affine hull $\operatorname{aff}(\Delta_c) = \{x \in \mathbb{R}^c \colon \langle x, \mathbb{1}_c \rangle = 1\}$, the *relative interior* $\operatorname{rint} \Delta_c$ of Δ_c is defined as the interior of Δ_c with respect to the subspace topology of $\operatorname{aff}(\Delta_c) \subseteq \mathbb{R}^c$. This set contains discrete distributions with full support

$$\mathcal{S}_c = \operatorname{rint} \Delta_c = \{p \in \mathbb{R}^c \colon p > 0, \quad \langle p, \mathbb{1}_c \rangle = 1\} \equiv \mathcal{P}([c]). \tag{2.3}$$

Information geometry is the study of sets like $\mathcal{S}_c$ as *statistical manifolds*. We will now successively equip $\mathcal{S}_c$ with structure, working up to this notion. For reference on information geometry, see [7, 12].

Lemma 2.1 ($\mathcal{S}_c$ is Topological Manifold) *$\mathcal{S}_c$ equipped with the subspace topology in* $\mathrm{aff}(\Delta_c)$ *is a topological manifold.*

Proof. The affine hull $\mathrm{aff}(\Delta_c)$ is a linear subspace of $\mathbb{R}^c$ which is homeomorphic to $\mathbb{R}^{c-1}$. $\mathcal{S}_c$ is an open subset of $\mathrm{aff}(\Delta_c)$ and thus, second-countability of its subspace topology and the Hausdorff property are inherited from $\mathbb{R}^{c-1}$. Through the same homeomorphism between $\mathrm{aff}(\Delta_c)$ and $\mathbb{R}^{c-1}$ restricted to $\mathcal{S}_c$, every point in $\mathcal{S}_c$ also has a neighborhood which is homeomorphic to an open subset of $\mathbb{R}^{c-1}$. □

There are two global charts for $\mathcal{S}_c$ which we will frequently return to in the following chapters.

Definition 2.2 (e-coordinates) *Let $\varphi_e \colon \mathcal{S}_c \to \mathbb{R}^{c-1}$ be the map*

$$p \mapsto \varphi_e(p) = \left(\log \frac{p_1}{p_c}, \dots, \log \frac{p_{c-1}}{p_c} \right)^\top \tag{2.4}$$

with inverse $\varphi_e^{-1} \colon \mathbb{R}^{c-1} \to \mathcal{S}_c$ defined by

$$\theta \mapsto \varphi_e^{-1}(\theta) = \exp\left(\begin{pmatrix} \theta \\ 0 \end{pmatrix} - \psi(\theta) \mathbb{1}_c \right) \tag{2.5a}$$

$$\psi(\theta) = \log(1 + \langle \exp(\theta), \mathbb{1}_{c-1} \rangle). \tag{2.5b}$$

ψ is called log-partition function*, φ_e is called exponential coordinate chart,* e-coordinate chart *for short. The coordinates $\theta \in \mathbb{R}^{c-1}$ are called exponential coordinates or* e-coordinates*.*

The right hand side of (2.5a) is the softmax function applied to $(\theta^\top, 0)^\top$. The motivation behind the term *exponential coordinates* is that $\mathcal{S}_c$ can be seen as an *exponential family* of distributions. This means that every $p \in \mathcal{S}_c$ has a unique representation of the form (2.5). In this context, θ are called *natural parameters*.

Definition 2.3 (m-coordinates) *Let $D_m = \{\mu \in \mathbb{R}^{c-1} \colon \mu > 0,\ \langle \mu, \mathbb{1}_{c-1} \rangle < 1\}$ and define $\varphi_m \colon \mathcal{S}_c \to D_m$ by*

$$p \mapsto \varphi_m(p) = (p_1, \dots, p_{c-1}) \tag{2.6}$$

with inverse $\varphi_m^{-1} \colon D_m \to \mathcal{S}_c$ given as

$$\mu \mapsto \varphi_m^{-1}(\mu) = (\mu^\top, 1 - \langle \mathbb{1}_{c-1}, \mu \rangle)^\top. \tag{2.7}$$

The map φ_m is called mixture-coordinate chart, m-coordinate chart *for short. The coordinates $\mu \in D_m$ are called mixture-coordinates or* m-coordinates*.*

The term *mixture coordinates* is motivated by the fact that $\mathcal{S}_c$ is a *mixture family* of distributions generated by the extremal points $\delta_{\{1\}}, \cdots, \delta_{\{c\}}$ corresponding to unit vectors in $\mathbb{R}^c$. This means that every point in $\mathcal{S}_c$ can be written uniquely as a mixture

$$\mathcal{S}_c \ni p = \sum_{j \in [c]} p_j \delta_{\{j\}}, \qquad p_j > 0, \quad \sum_{j \in [c]} p_j = 1. \tag{2.8}$$

In addition, mixtures of distributions are convex combinations in m-coordinates.

The collection $\{\varphi_e, \varphi_m\}$ defines a *smooth atlas* for $\mathcal{S}_c$, because both transition maps $\varphi_e \circ \varphi_m^{-1}$ and $\varphi_m \circ \varphi_e^{-1}$ are smooth as functions between open subsets of $\mathbb{R}^{c-1}$. The smooth structure determined by this atlas turns $\mathcal{S}_c$ into a *smooth manifold*.

In m-coordinates, a basis for the tangent space at any point $p = \varphi_m^{-1}(\mu) \in \mathcal{S}_c$ is defined by

$$\Big\{\frac{\partial}{\partial \mu^i}\Big|_p\Big\}_{i\in[c-1]}, \qquad \frac{\partial}{\partial \mu^i}\varphi_m(p)^j = \delta_i^j, \quad i,j \in [c-1]. \tag{2.9}$$

The dual basis $\{d\mu^i|_p\}_{i\in[c-1]}$ of the cotangent space at p satisfies

$$d\mu^i|_p\Big(\frac{\partial}{\partial \mu^j}\Big|_p\Big) = \delta_j^i. \tag{2.10}$$

In the following chapters, we will usually work with expressions in the ambient space $\mathbb{R}^c$. To clarify this, regard φ_m^{-1} as a map $\mathbb{R}^{c-1} \to \mathbb{R}^c$ and consider the smooth curve $t \mapsto \varphi_m^{-1}(\mu + t\xi)$ for some $\mu, \xi \in \mathbb{R}^{c-1}$. Define the tangent vector

$$v = \frac{d}{dt}\varphi_m^{-1}(\mu + t\xi)|_{t=0} = (\xi^\top, -\langle \xi, \mathbb{1}_{c-1}\rangle)^\top \in T_p\mathbb{R}^c \equiv \mathbb{R}^c \tag{2.11}$$

along this curve at $p = \varphi_m^{-1}(\mu)$. (2.11) associates every $\xi \in \mathbb{R}^{c-1}$ with a unique $v \in T_p\mathbb{R}^c$ subject to $\langle v, \mathbb{1}_c\rangle = 0$. This allows to identify the linear subspace

$$T_0\mathcal{S}_c = \{v \in \mathbb{R}^c \colon \langle v, \mathbb{1}_c\rangle = 0\} \tag{2.12}$$

of $\mathbb{R}^c \equiv T_p\mathbb{R}^c$ with the *tangent space* of $\mathcal{S}_c$ at p. More precisely, $v \in T_0\mathcal{S}_c$ is identified with the tangent vector

$$v \equiv \sum_{i\in[c-1]} v^i \frac{\partial}{\partial \mu^i}\Big|_p. \tag{2.13}$$

Using this identification, $\mathcal{S}_c$ has the trivial tangent bundle

$$T\mathcal{S}_c \cong \mathcal{S}_c \times T_0\mathcal{S}_c. \tag{2.14}$$

We turn to the construction of a Riemannian metric on $\mathcal{S}_c$. Define the *log-likelihood* vectors

$$\ell(\mu) = \log \varphi_m^{-1}(\mu) \in \mathbb{R}^c, \qquad \mu \in D_m \tag{2.15}$$

and the *score vectors*

$$\partial_j \ell(\mu) = \frac{\partial}{\partial \mu^j}\ell(\mu), \qquad \mu \in D_m,\ j \in [c-1]. \tag{2.16}$$

The *Fisher information matrix* in m-coordinates is the $(c-1)\times(c-1)$ matrix with entries

$$g_{ij} = \mathbb{E}_{\varphi_m^{-1}(\mu)}[\partial_i\ell(\mu)\partial_j\ell(\mu)] = \sum_{l\in[c]} \varphi_m^{-1}(\mu)^l \partial_i\ell(\mu)^l \partial_j\ell(\mu)^l \tag{2.17a}$$

$$= \delta_{ij}\frac{1}{\mu^i} + \frac{1}{1 - \langle \mu, \mathbb{1}_{c-1}\rangle}, \qquad i,j \in [c-1]. \tag{2.17b}$$

We can use this matrix to define a symmetric covariant tensor on $\mathcal{S}_c$, called *Fisher-Rao metric tensor* $g_p\colon T_0\mathcal{S}_c \times T_0\mathcal{S}_c \to \mathbb{R}$ which acts on a pair of tangent vectors at p by

$$g_p(u,v) \overset{(2.13)}{=} \sum_{(i,j)\in[c-1]^2} g_{ij} d\mu^i\Big(\sum_{l\in[c-1]} u^l \frac{\partial}{\partial \mu^l}\Big|_p\Big) d\mu^j\Big(\sum_{l\in[c-1]} v^l \frac{\partial}{\partial \mu^l}\Big|_p\Big) \tag{2.18a}$$

$$\overset{(2.10)}{=} \sum_{(i,j)\in[c-1]^2} g_{ij} u^i v^j \tag{2.18b}$$

$$\overset{(2.17)}{=} \sum_{(i,j)\in[c-1]^2} \Big(\delta_{ij}\frac{1}{\mu^i} + \frac{1}{1-\langle \mu, \mathbb{1}_{c-1}\rangle}\Big) u^i v^j \tag{2.18c}$$

$$= \frac{u^c v^c}{1-\langle \mu, \mathbb{1}_{c-1}\rangle} + \sum_{i\in[c-1]} \frac{u^i v^i}{\mu^i} \tag{2.18d}$$

$$= \Big\langle \frac{u}{p}, v\Big\rangle, \qquad p = \varphi_m^{-1}(\mu). \tag{2.18e}$$

In (2.18d), we have used that, due to the linear constraint in (2.12), the last entry of v (resp. u) can be written as

$$v^c = -\sum_{i\in[c-1]} v^i. \tag{2.19}$$

Because the Fisher information matrix is symmetric and positive definite, the action on a pair of tangent vectors (2.18) actually defines a Riemannian metric, called *Fisher-Rao metric* on $T_0\mathcal{S}_c$.

Definition 2.4 (Fisher-Rao Metric) *On the tangent space $T_0\mathcal{S}_c$ at every $p \in \mathcal{S}_c$, the* Fisher-Rao metric *is defined by the inner product*

$$\langle u, v\rangle_p = g_p(u,v) = \Big\langle \frac{u}{p}, v\Big\rangle, \qquad u, v \in T_0\mathcal{S}_c. \tag{2.20}$$

Note, that the Fisher-Rao metric has a coordinate-independent representation as

$$\langle u, v\rangle_p = \mathbb{E}_p[(u\ell)(v\ell)] \tag{2.21}$$

where the action of tangent vectors on ℓ is the directional derivative of each real-valued log-likelihood component function.

Let $\{e_1, \dots, e_c\}$ denote the standard basis of $\mathbb{R}^c$ and denote the tangent coordinate basis of $T_p\mathbb{R}^c$ by

$$\Big\{\frac{\partial}{\partial x^1}, \dots, \frac{\partial}{\partial x^c}\Big\}. \tag{2.22}$$

Then $\{e_1, \dots, e_{c-1}\}$ is a basis of the linear subspace $\operatorname{aff}(\Delta_c)$ and

$$\Big\{\frac{\partial}{\partial x^1}, \dots, \frac{\partial}{\partial x^{c-1}}\Big\} \tag{2.23}$$

can be identified with a basis of $T_p \operatorname{aff}(\Delta_c)$ for any $p \in \operatorname{aff}(\Delta_c)$. Since $\mathcal{S}_c$ is an open submanifold of $\operatorname{aff}(\Delta_c)$, the differential of the inclusion map $d\imath\colon T_0\mathcal{S}_c \to T_p \operatorname{aff}(\Delta_c)$ is an

isomorphism at any $p \in \mathcal{S}_c$. This identifies (2.23) with a basis of $T_0\mathcal{S}_c$. In light of (2.6), components in this basis coincide with components in m-coordinates.

Let $f\colon \mathbb{R}^c \to \mathbb{R}$ be a smooth function with restriction $\widetilde{f}$ to $\mathcal{S}_c$. Denote the gradient of f as the covector field

$$df(p) = \sum_{i\in[c]} \partial_i f(p) dx^i|_p \tag{2.24}$$

and collect the components of (2.24) in the vector

$$\partial f(p) = (\partial_1 f(p), \dots, \partial_c f(p)). \tag{2.25}$$

The action of (2.24) on a tangent vector $v \in T_p\mathbb{R}^c$ which lies in the subspace $T_0\mathcal{S}_c$ reads

$$df(p)[v] \overset{(2.13)}{=} \sum_{i\in[c]} \partial_i f(p) dx^i|_p \Big(\sum_{j\in[c]} v^j \frac{\partial}{\partial x^j}\Big|_p \Big) = \sum_{i\in[c]} \partial_i f(p) v^i \tag{2.26a}$$

$$\overset{(2.19)}{=} \sum_{i\in[c-1]} \partial_i f(p) v^i - \partial_c f(p) \sum_{i\in[c-1]} v^i \tag{2.26b}$$

$$= \sum_{i\in[c-1]} (\partial_i f(p) - \partial_c f(p)) v^i. \tag{2.26c}$$

Because v is also tangent to $\mathcal{S}_c$, we have $d\widetilde{f}(p)[v] = df(p)[v]$ for any $p \in \mathcal{S}_c$. By (2.18b), the inner product (2.20) with a second tangent vector $u \in T_0\mathcal{S}_c$ reads

$$\langle u, v\rangle_p = \sum_{(i,j)\in[c-1]^2} g_{ij} u^i v^j \tag{2.27}$$

for components u^i, v^j, $i, j \in [c-1]$ in m-coordinates. As outlined above, these components of v coincide with the ones in (2.26). The *Riemannian gradient* of $\widetilde{f}$ at p is defined as the vector u such that $d\widetilde{f}(p)[v]$ matches (2.27) for every $v \in T_0\mathcal{S}_c$. Comparing (2.26) with (2.27), the components of u in m-coordinates are found by solving a full-rank linear system. Inversion of the Fisher information matrix (2.17) gives

$$g^{-1} = \mathrm{Diag}(\mu) - \mu\mu^\top \tag{2.28}$$

and we consequently find the Riemannian gradient components ($i \in [c-1]$)

$$u^i = \sum_{j\in[c-1]} g^{-1}_{ij} (\partial_j f(p) - \partial_c f(p)) \tag{2.29a}$$

$$= \sum_{j\in[c-1]} (\delta_{ij}\mu^i - \mu^i\mu^j)(\partial_j f(p) - \partial_c f(p)) \tag{2.29b}$$

$$= \mu^i(\partial_i f(p) - \partial_c f(p)) - \mu^i \sum_{j\in[c-1]} \mu^j (\partial_j f(p) - \partial_c f(p)) \tag{2.29c}$$

$$= \mu^i(\partial_i f(p) - \partial_c f(p)) - \mu^i \sum_{j\in[c-1]} \mu^j \partial_j f(p) + \mu^i \partial_c f(p)(1 - p_c) \tag{2.29d}$$

$$= \mu^i \partial_i f(p) - \mu^i \langle \partial f(p), p\rangle. \tag{2.29e}$$

Since

$$u^c = -\sum_{i\in[c-1]} u^i = \sum_{i\in[c-1]} \mu^i\langle\partial f(p), p\rangle - \mu^i\partial_i f(p) \tag{2.30a}$$
$$= \langle\partial f(p), p\rangle(1-p_c) - \sum_{i\in[c-1]} \mu^i\partial_i f(p) \tag{2.30b}$$
$$= \langle\partial f(p), p\rangle(1-p_c) - (\langle\partial f(p), p\rangle - p_c\partial_c f(p)) \tag{2.30c}$$
$$= p_c\partial_c f(p) - p_c\langle\partial f(p), p\rangle \tag{2.30d}$$

we can compute $u \in T_0\mathcal{S}_c$ in the simple vectorized fashion

$$u = p \diamond \partial f(p) - \langle\partial f(p), p\rangle\partial f(p). \tag{2.31}$$

The expression (2.31) of Riemannian gradients in terms of an ambient function gradient occurs frequently in the following chapters.

Definition 2.5 (Replicator Operator) *For any $p \in \mathcal{S}_c \subseteq \mathbb{R}^c$, the linear operator $R_p\colon \mathbb{R}^c \to T_0\mathcal{S}_c$ defined by*

$$R_p[v] = (\mathrm{Diag}(p) - pp^\top)v = p \diamond v - \langle p, v\rangle p, \qquad v \in \mathbb{R}^c \tag{2.32}$$

is called replicator operator.

In Section 3.2, the replicator operator will be derived in an alternate way, based on principled modelling of population dynamics. Here, we collect some basic properties. Let $\Pi_0\colon \mathbb{R}^c \to T_0\mathcal{S}_c$ denote the projection

$$\Pi_0 v = v - \frac{1}{c}\langle v, \mathbb{1}_c\rangle\mathbb{1}_c. \tag{2.33}$$

One easily verifies that

$$R_p \circ \Pi_0 = \Pi_0 \circ R_p = R_p, \qquad \forall p \in \mathcal{S}_c. \tag{2.34}$$

Thus, (2.31) can equivalently be seen as $\partial f(p)$ being projected to $T_0\mathcal{S}_c$ before applying R_p. (2.34) also implies that the image of the replicator operator is indeed a subset of $T_0\mathcal{S}_c$.

Lemma 2.6 (Inverse of Replicator) *For any $p \in \mathcal{S}_c$, the replicator operator R_p restricted to $T_0\mathcal{S}_c$ is a bijection on $T_0\mathcal{S}_c$ with inverse defined by*

$$R_p^{-1}[u] = \Pi_0\frac{u}{p}. \tag{2.35}$$

Proof. Let $v \in T_0\mathcal{S}_c$ be in the kernel of R_p. Then $0 = R_p[v] = p \diamond v - \langle p, v\rangle p$ implies $v = \langle p, v\rangle\mathbb{1}_c$ and, due to $v \in T_0\mathcal{S}_c$, this gives

$$v = \Pi_0 v = \langle p, v\rangle\Pi_0\mathbb{1}_c = 0 \in T_0\mathcal{S}_c. \tag{2.36}$$

We directly show the shape of the inverse replicator (2.35). Let $v \in T_0\mathcal{S}_c$, then

$$(R_p^{-1} \circ R_p)[v] = \Pi_0\frac{\mathbb{1}_c}{p} \diamond R_p[v] = \Pi_0(v - \langle p, v\rangle\mathbb{1}_c) = \Pi_0 v = v. \tag{2.37}$$

□

A central construction in *information geometry* evolves around two affine connections on $\mathcal{S}_c$, called m-connection $\nabla^{(m)}$ and e-connection $\nabla^{(e)}$. At $p \in \mathcal{S}_c$ with m-coordinates $\varphi_m(p) = \mu$, these connections are defined by the Christoffel symbols

$$\langle \nabla^{(e)}_{\partial_i} \partial_j, \partial_k \rangle_g = \Gamma^{(e)}_{ij,k}, \qquad \left(\Gamma^{(e)}_{ij,k}\right)_\mu = \mathbb{E}_p\Big[\partial_i\partial_j\ell_\mu \partial_k \ell_\mu\Big] \tag{2.38a}$$

$$\langle \nabla^{(m)}_{\partial_i} \partial_j, \partial_k \rangle_g = \Gamma^{(m)}_{ij,k}, \qquad \left(\Gamma^{(m)}_{ij,k}\right)_\mu = \mathbb{E}_p\Big[\Big(\partial_i\partial_j\ell_\mu + \partial_i\ell_\mu\partial_j\ell_m u\Big)\partial_k\ell_\mu\Big] \tag{2.38b}$$

for $i, j, k \in [c-1]$ using the shorthand $\partial_i = \frac{\partial}{\partial \mu^i}$. These connections are constructed such that m-coordinates and e-coordinates are affine coordinates for m-connection and e-connection respectively [7, Theorem 2.4]. We call the respective geodesic curves m-geodesics and e-geodesics. In light of (2.7), m-geodesics $t \mapsto \gamma_m(t)$ coincide with affine curves in $\mathbb{R}^c$ which intersect $\mathcal{S}_c$ on an open subset. In particular, these curves are not defined in $\mathcal{S}_c$ for all times $t > 0$. In contrast, e-geodesics are defined in $\mathcal{S}_c$ for all times.

Lemma 2.7 (e-Geodesic Curves) *Let $t \mapsto \gamma_e(t)$ be the e-geodesic curve with $\gamma(0) = p \in \mathcal{S}_c$ and $\dot\gamma_e(0) = v \in T_0\mathcal{S}_c$. Then*

$$\gamma(1) = \frac{p \diamond \exp \frac{v}{p}}{\left\langle p, \exp \frac{v}{p} \right\rangle} \in \mathcal{S}_c. \tag{2.39}$$

Proof. Let $\mu = \varphi_m(p)$ and $v = (\xi^\top, -\langle \xi, \mathbb{1}_{c-1}\rangle)^\top$ for $\xi \in T_\mu \mathbb{R}^{c-1} \equiv \mathbb{R}^{c-1}$. We first transform μ into e-coordinates θ_0 and ξ into a tangent vector η in the e-coordinate system. A straightforward computation shows

$$\theta_0 = (\varphi_e \circ \varphi_m^{-1})(\mu) = \Big(\log \frac{\mu_1}{p_c}, \dots, \frac{\mu_{c-1}}{p_c}\Big) \tag{2.40a}$$

$$\eta = d(\varphi_e \circ \varphi_m^{-1})(\mu)[\xi] = \frac{\xi}{\mu} - \frac{v_c}{p_c}\mathbb{1}_{c-1}. \tag{2.40b}$$

Let $\theta_1 = \theta_0 + \eta$ such that $\gamma(1) = \varphi_e^{-1}(\theta_1)$. The partition function reads

$$\exp(\psi(\theta_1)) = 1 + \langle \exp\theta_1, \mathbb{1}_{c-1}\rangle = 1 + \Big\langle \exp\Big(\frac{\xi}{\mu} - \frac{v_c}{p_c}\mathbb{1}_{c-1}\Big), \frac{\mu}{p_c}\Big\rangle \tag{2.41a}$$

$$= \frac{1}{p_c}\exp\Big(-\frac{v_c}{p_c}\Big)\Big(p_c \exp\Big(\frac{v_c}{p_c}\Big) + \Big\langle \mu, \exp\frac{\xi}{\mu}\Big\rangle\Big) \tag{2.41b}$$

$$= \frac{1}{p_c}\exp\Big(-\frac{v_c}{p_c}\Big)\Big\langle p, \exp\frac{v}{p}\Big\rangle. \tag{2.41c}$$

Using (2.5) and (2.41), we find the first $c-1$ components of $\gamma(1)$ as

$$\exp(-\psi(\theta_1))\exp(\theta_1) = \frac{p_c \exp\left(\frac{v_c}{p_c}\right)}{\left\langle p, \exp\frac{v}{p}\right\rangle}\frac{1}{p_c}\mu \diamond \exp\Big(\frac{\xi}{\mu} - \frac{v_c}{p_c}\mathbb{1}_{c-1}\Big) = \frac{\mu \diamond \exp\left(\frac{\xi}{\mu}\right)}{\left\langle p, \exp\frac{v}{p}\right\rangle} \tag{2.42}$$

and the last component as

$$\exp(-\psi(\theta_1)) = \frac{p_c \exp\left(\frac{v_c}{p_c}\right)}{\left\langle p, \exp\frac{v}{p}\right\rangle}. \tag{2.43}$$

□

In particular, for every $v \in T_0\mathcal{S}_c$ and every $p \in \mathcal{S}_c$, the endpoint $\gamma_e(1)$ is a well-defined point in $\mathcal{S}_c$. Lemma 2.7 specifies the *exponential map* relative to the e-connection as

$$\operatorname{Exp}_p \colon T_0\mathcal{S}_c \to \mathcal{S}_c, \qquad v \mapsto \operatorname{Exp}_p = \frac{p \diamond \exp \frac{v}{p}}{\left\langle p, \exp \frac{v}{p} \right\rangle}. \tag{2.44}$$

The global existence of e-geodesic curves is a useful property with regard to numerical implementation of algorithms working with state in $\mathcal{S}_c$. In this context, we will frequently use the following map.

Definition 2.8 (Lifting Map) *For $p \in \mathcal{S}_c$ and $v \in T_0\mathcal{S}_c$, the* lifting map *is defined by*

$$\exp_p \colon T_0\mathcal{S}_c \to \mathcal{S}_c, \qquad \exp_p(v) = (\operatorname{Exp}_p \circ R_p)(v) = \frac{p \diamond \exp(v)}{\langle p, \exp(v) \rangle}. \tag{2.45}$$

We study basic properties of the lifting map, which will be used repeatedly in the following chapters.

Lemma 2.9 (Differential of the Lifting Map) *For $p \in \mathcal{S}_c$ and $v \in T_0\mathcal{S}_c$, the lifting map has differential*

$$\mathrm{d} \exp_p(v) = R_{\exp_p(v)}. \tag{2.46}$$

Proof. Let $\gamma \colon \mathbb{R} \to T_0\mathcal{S}_c$ be a smooth curve with $\gamma(0) = v$ and $\dot{\gamma}(0) = u$. Then

$$\begin{aligned}
\mathrm{d}\big(\exp_p(v)\big)\,[u] &= \frac{\mathrm{d}}{\mathrm{d}t}\big(\exp_p(\gamma(t))\big)|_{t=0} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{p \diamond \exp(\gamma(t))}{\langle p, \exp(\gamma(t)) \rangle}\right)|_{t=0} && \text{(2.47a)}\\
&= \frac{p \diamond \exp(v)}{\langle p, \exp(v) \rangle} \diamond u - \frac{p \diamond \exp(v) \diamond (\langle p \diamond \exp(v), u \rangle \mathbb{1}_c)}{\langle p, \exp(v) \rangle^2} && \text{(2.47b)}\\
&= \exp_p(v) \diamond u - \exp_p(v) \diamond \left(\frac{\langle p \diamond \exp(v), u \rangle}{\langle p, \exp(v) \rangle} \mathbb{1}_c\right) && \text{(2.47c)}\\
&= \exp_p(v) \diamond u - \exp_p(v) \diamond (\langle \exp_p(v), u \rangle \mathbb{1}_c) && \text{(2.47d)}\\
&= R_{\exp_p(v)}[u]. && \text{(2.47e)}
\end{aligned}$$

□

Lemma 2.10 (Action Property) *For $p \in \mathcal{S}_c$ and $u, v \in T_0\mathcal{S}_c$, the lifting map has the* action property

$$\exp_p(u + v) = \exp_{\exp_p(u)}(v), \qquad p \in \mathcal{S}_c, \quad u, v \in T_0\mathcal{S}_c. \tag{2.48}$$

Proof. We directly compute

$$\exp_p(u + v) = \frac{(p \diamond \exp(u)) \diamond \exp(v)}{\langle p \diamond \exp(u), \exp(v) \rangle} = \frac{\frac{p \diamond \exp(u)}{\langle p, \exp(u) \rangle} \diamond \exp(v)}{\left\langle \frac{p \diamond \exp(u)}{\langle p, \exp(u) \rangle}, \exp(v) \right\rangle} = \exp_{\exp_p(u)}(v). \tag{2.49}$$

□

As a function $\mathbb{R}^c \to T_0\mathcal{S}_c$, the lifting map has an invariance under vectors which are orthogonal to $T_0\mathcal{S}_c$. For any $v \in T_0\mathcal{S}_c$, $p \in \mathcal{S}_c$ and any $\alpha \in \mathbb{R}$, we find

$$\exp_p(v + \alpha\mathbb{1}_c) = \frac{p \diamond \exp v \diamond (\exp(\alpha)\mathbb{1}_c)}{\langle p, \exp v \diamond (\exp(\alpha)\mathbb{1}_c)\rangle} = \frac{\exp(\alpha)(p \diamond \exp v)}{\exp(\alpha)\langle p, \exp v\rangle} = \exp_p(v). \tag{2.50}$$

In particular, comparison with (2.33) shows

$$\exp_p = \exp_p \circ \Pi_0, \tag{2.51}$$

which allows to extend the domain of $\exp_p$ to the entirety of $\mathbb{R}^c$. (2.50) is a useful property for numerical implementation, because it allows to shift all entries of v such that the largest is zero, avoiding overflow when evaluating the exponential function. Another useful property of the lifting map for implementation purposes is its relation to the softmax function

$$\operatorname{softmax}\colon \mathbb{R}^c \to \mathcal{S}_c, \qquad v \mapsto \operatorname{softmax}(v) = \frac{\exp(v)}{\langle \mathbb{1}_c, \exp(v)\rangle} \tag{2.52}$$

with numerically stable implementation available in many software libraries. The lifting map can be written in terms of the softmax function as

$$\exp_p(v) = \frac{p \diamond \exp(v)}{\langle \mathbb{1}_c, p \diamond \exp(v)\rangle} = \frac{\exp(v + \log p)}{\langle \mathbb{1}_c, \exp(v + \log p)\rangle} = \operatorname{softmax}(v + \log p). \tag{2.53}$$

A simple special case is lifting at the barycenter

$$\mathbb{1}_{\mathcal{S}} = \frac{1}{c}\mathbb{1}_c \in \mathcal{S}_c \tag{2.54}$$

of $\mathcal{S}_c$ which simply reads

$$\exp_{\mathbb{1}_{\mathcal{S}}}(v) = \frac{\frac{1}{c}\mathbb{1}_c \diamond \exp(v)}{\frac{1}{c}\langle \mathbb{1}_c, \exp(v)\rangle} = \operatorname{softmax}(v). \tag{2.55a}$$

The e-exponential map lifts tangent vectors $v \in T_0\mathcal{S}_c$ at p to simplex points $\operatorname{Exp}_p(v)$. From (2.44), it is clear that $\operatorname{Exp}_p(v)$ is well-defined and lies in $\mathcal{S}_c$ for every $v \in T_0\mathcal{S}_c$. In addition, $\operatorname{Exp}_p\colon T_0\mathcal{S}_c \to \mathcal{S}_c$ is bijective for every $p \in \mathcal{S}_c$ with inverse

$$\operatorname{Exp}_p^{-1}\colon \mathcal{S} \to T_0\mathcal{S}_c, \qquad \widetilde{p} \mapsto \operatorname{Exp}_p^{-1}(\widetilde{p}) = R_p \log \frac{\widetilde{p}}{p}, \tag{2.56}$$

which we readily verify by

$$R_p \log \frac{\mathbb{1}_c}{p} \diamond \operatorname{Exp}_p(v) = R_p\Big(\frac{v}{p} - \log\Big\langle p, \exp\frac{v}{p}\Big\rangle \mathbb{1}_c\Big) \overset{(2.34)}{=} R_p \frac{v}{p} = v - \underbrace{\langle \mathbb{1}_c, v\rangle}_{=0} \mathbb{1}_c \tag{2.57}$$

for every $v \in T_0\mathcal{S}_c$. Since the replicator operator is bijective on $T_0\mathcal{S}_c$ by Lemma 2.6, the lifting map $\exp_p = \operatorname{Exp}_p \circ R_p$ is also a bijection and (2.56) allows to easily compute its inverse

$$\exp_p^{-1}\colon \mathcal{S}_c \to T_0\mathcal{S}_c, \qquad \widetilde{p} \mapsto \exp_p^{-1}(\widetilde{p}) = \Pi_0 \log \frac{\widetilde{p}}{p} \tag{2.58}$$

due to

$$\exp_p^{-1}(\widetilde{p}) = (R_p^{-1} \circ \operatorname{Exp}_p^{-1})(\widetilde{p}) \overset{(2.35),(2.56)}{=} \Pi_0 \frac{\mathbb{1}_c}{p} R_p \log \frac{\widetilde{p}}{p} = \Pi_0 \log \frac{\widetilde{p}}{p} \tag{2.59}$$

In light of (2.55), this also shows that the inverse of the softmax function on $T_0\mathcal{S}_c$ reads

$$\operatorname{softmax}^{-1} \colon \mathcal{S}_c \to T_0\mathcal{S}_c, \qquad \widetilde{p} \mapsto \operatorname{softmax}^{-1}(\widetilde{p}) = \Pi_0 \log \widetilde{p}. \tag{2.60}$$

2.2 Statistical Learning Theory and Deep Learning

2.2.1 Risk Certification

In the second part of this thesis, we will work on statistical aspects of structured prediction. The central goal is developing a statistical learning theory which assesses model generalization through *tight and computable risk certificates*. The following introduction to the topic is based mainly on [3]. Let $\ell \colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ be a loss function which compares elements of a label space $\mathcal{Y}$. We will focus on classification where $\mathcal{Y} = [c]$ is a discrete set of classes and the natural loss function is 01 loss

$$\ell^{01}(y_1, y_2) = \begin{cases} 0, & \text{if } y_1 = y_2 \\ 1, & \text{else.} \end{cases} \tag{2.61}$$

Let $\mathcal{X}$ be a metric space and suppose we have access to a sample of m independently drawn data from an unknown distribution μ on $\mathcal{X} \times \mathcal{Y}$

$$Z = ((X_1, Y_1), \ldots, (X_m, Y_m)) \sim \mu^m. \tag{2.62}$$

For a predictor $\phi \colon \mathcal{X} \to \mathcal{Y}$, mean loss over the sample is called the *empirical risk* of ϕ

$$\mathcal{R}_m(\phi) = \frac{1}{m} \sum_{i \in [m]} \ell(\phi(X_i), Y_i). \tag{2.63}$$

In classification, the empirical risk with respect to 01 loss is the error rate of ϕ on the sample. Empirical risk is a tractable surrogate for the intractable *risk* of ϕ

$$\mathcal{R}(\phi) = \mathbb{E}_{(X,Y)\sim\mu}[\ell(\phi(X), Y)] \tag{2.64}$$

which measures the performance of ϕ on any data drawn from the unknown distribution μ, including *out-of-sample* data. Statistical learning is the task of finding predictors with low risk. Since the actual model risk is intractable, this is naturally framed as a predictor with low empirical risk which also generalizes well, i.e. has small *generalization gap*

$$\mathcal{R}(\phi) - \mathcal{R}_m(\phi). \tag{2.65}$$

A first observation to this end is that risk is the expected value of empirical risk under the draw of the sample

$$\mathbb{E}_{Z\sim\mu^m} \mathcal{R}_m(\phi) = \frac{1}{m} \sum_{i \in [m]} \mathbb{E}_{(X_i,Y_i)\sim\mu} \ell(\phi(X_i), Y_i) = \frac{1}{m} \sum_{i \in [m]} \mathcal{R}(\phi) = \mathcal{R}(\phi). \tag{2.66}$$

This perspective allows us to view generalization in terms of empirical risk concentrating on its mean when more independent data are collected. Loosely speaking, the underlying *concentration of measure phenomenon* can be summarized as

> *Stable* functions of a *large number of independent random variables* concentrate on their mean.

A simple case of concentration of measure is the *law of large numbers*. If many independent random variables are given, all following the same probability distribution, their sample mean converges to the expected value. Under additional stability assumptions, *concentration inequalities* can make statements about the rate of convergence by bounding the probability of deviation from the mean. A prominent example is Hoeffding's inequality.

Theorem 2.11 (Hoeffding's inequality [93]) *Let $U_1, \dots, U_m$ be independent random variables, each almost surely taking values in $[a, b]$. For any $\tau > 0$ it holds*

$$\mathbb{E}\Big[\exp\Big(\frac{\tau}{m}\sum_{i\in[m]} U_i - \mathbb{E}[U_i]\Big)\Big] \leq \exp\Big(\frac{\tau^2(b-a)^2}{8m}\Big) \tag{2.67}$$

as well as

$$\mathbb{P}\Big(\frac{1}{m}\sum_{i\in[m]} U_i - \mathbb{E}[U_i] \geq \tau\Big) \leq \exp\Big(-\frac{2\tau^2 m}{(b-a)^2}\Big). \tag{2.68}$$

Returning to the question of generalization, assume that the loss ℓ is bounded, i.e. $\ell \in [0, C]$ for some constant C. This is naturally the case in classification because $\ell^{01} \in [0, 1]$. Define the independent random variables

$$U_i = \mathcal{R}(\phi) - \ell(\phi(X_i), Y_i), \qquad i \in [m] \tag{2.69}$$

with distribution $\widetilde{\mu}$ and mean 0. Each U_i takes values in the interval $[\mathcal{R}(\phi) - C, \mathcal{R}(\phi)]$. Hoeffding's inequality (2.68) thus gives

$$\begin{aligned}
\mathbb{P}_{Z\sim\mu^m}\Big(\mathcal{R}(\phi) - \mathcal{R}_m(\phi) > \tau\Big) &= \mathbb{P}_{Z\sim\mu^m}\Big(\mathcal{R}(\phi) - \frac{1}{m}\sum_{i\in[m]} \ell(\phi(X_i), Y_i) > \tau\Big) && \text{(2.70a)}\\
&= \mathbb{P}_{Z\sim\mu^m}\Big(\frac{1}{m}\sum_{i\in[m]} \mathcal{R}(\phi) - \ell(\phi(X_i), Y_i) > \tau\Big) && \text{(2.70b)}\\
&= \mathbb{P}_{U\sim\widetilde{\mu}^m}\Big(\frac{1}{m}\sum_{i\in[m]} U_i > \tau\Big) && \text{(2.70c)}\\
&\leq \exp\Big(-\frac{2\tau^2 m}{C^2}\Big) && \text{(2.70d)}
\end{aligned}$$

To estimate the generalization gap, we can further re-write (2.70) by fixing an error probability

$$\delta = \exp\Big(-\frac{2m\tau^2}{C^2}\Big) \tag{2.71}$$

instead of the margin τ, which results in

$$\mathcal{R}(\phi) \leq \mathcal{R}_m(\phi) + C\sqrt{\frac{\log \frac{1}{\delta}}{2m}} \tag{2.72}$$

with probability at least $1 - \delta$ over the draw of the sample. At this point, we have achieved a bound on the true, intractable risk $\mathcal{R}(\phi)$, which is easily computable, holds with arbitrarily high probability, and tightens with rate $\mathcal{O}(\sqrt{m})$ as more data are collected. There were only two assumptions made

- boundedness of the loss function and
- independence of the sample.

In particular, we made no assumption on the distribution of data μ and we did not require differentiability of the loss function. The biggest problem with this bound is that it is only valid for a *single predictor ϕ, fixed before observing the data Z*. At the core of machine learning is the use of data to find a good predictor, for example through empirical risk minimization. However, since ϕ needs to be fixed before observing the data, (2.72) does not apply to predictors which are functions of the data. To illustrate this crucial point further, consider again the random variables $U_i = \mathcal{R}(\phi) - \ell(\phi(X_i), Y_i)$ in (2.69) and suppose

$$\phi = \phi^{\mathrm{ERM}} = \arg\min_{\phi \in \mathcal{H}} \mathcal{R}_m(\phi) \tag{2.73}$$

is the empirical risk minimizer. The conditional distribution of U_2 given any fixed value of $U_1 = u$ is no longer the marginal distribution of U_2. Changing the condition to $U_1 = u' \neq u$ by changing the corresponding value of $Z_1 = (X_1, Y_1)$ also changes ϕ^{ERM} which in turn changes the distribution of U_2. Thus, the variables U_1 and U_2 are not independent because ϕ^{ERM} is a function of the data. One way to tackle this problem is through a careful study of the resulting dependency. This is pursued within the *hypothesis stability* framework [55, 105, 33].

Here, we focus on a different established approach, leveraging *uniform convergence*. Define a *hypothesis class* $\mathcal{H}$ of functions $\phi_{\mathfrak{p}} \colon \mathcal{X} \to \mathcal{Y}$ parameterized by $\mathfrak{p} \in \Theta$. We will identify the hypothesis class $\mathcal{H}$ with the parameter space Θ through this parameterization and use the symbols $\phi_{\mathfrak{p}}$ and $\mathfrak{p}$ interchangeably to denote hypotheses, i.e. $\mathcal{R}(\phi_{\mathfrak{p}}) = \mathcal{R}(\mathfrak{p})$. If we can expand the bound (2.70) to hold simultaneously for all hypotheses in $\mathcal{H}$, this allows to bound the generalization gap of $\widehat{\phi}$ selected as a function of the data because all $\widehat{\phi} \in \mathcal{H}$ satisfy

$$\mathcal{R}(\widehat{\phi}) - \mathcal{R}_m(\widehat{\phi}) \leq \sup_{\phi \in \mathcal{H}} \mathcal{R}(\phi) - \mathcal{R}_m(\phi). \tag{2.74}$$

This idea of *uniform* bounds over the hypothesis class is at the core of multiple distinct approaches to statistical learning theory. In the simplest case, $\mathcal{H}$ only contains a finite number M of hypotheses and (2.70) can be made uniform by a union bound.

Lemma 2.12 (Union bound) *Given a finite number of random variables $U_1, \ldots, U_M$, it holds*

$$\mathbb{P}\Big(\sup_{i \in [M]} U_i > \tau\Big) = \mathbb{P}\Big(\bigcup_{i \in [M]} \{U_i > \tau\}\Big) \leq \sum_{i \in [M]} \mathbb{P}(U_i > \tau). \tag{2.75}$$

Returning to (2.70), Lemma 2.12 implies

$$\mathbb{P}_{Z\sim\mu^m}\Big(\sup_{\phi\in\mathcal{H}}\mathcal{R}(\phi)-\mathcal{R}_m(\phi)>\tau\Big)=\mathbb{P}_{Z\sim\mu^m}\Big(\bigcup_{\phi\in\mathcal{H}}\Big\{\mathcal{R}(\phi)-\mathcal{R}_m(\phi)>\tau\Big\}\Big) \tag{2.76a}$$

$$\leq\sum_{\phi\in\mathcal{H}}\mathbb{P}_{Z\sim\mu^m}\Big(\mathcal{R}(\phi)-\mathcal{R}_m(\phi)>\tau\Big) \tag{2.76b}$$

$$\leq M\exp\Big(-\frac{2\tau^2 m}{C^2}\Big) \tag{2.76c}$$

which can be viewed as the statement

$$\sup_{\phi\in\mathcal{H}}\mathcal{R}(\phi)-\mathcal{R}_m(\phi)\leq C\sqrt{\frac{\log\frac{M}{\delta}}{2m}} \tag{2.77}$$

with probability at least $1-\delta$ over the draw of the sample, analogous to (2.72). As is apparent from (2.77), this approach does not work for infinite hypothesis classes. However, there are different complexity measures for the hypothesis class $\mathcal{H}$ which induce uniform bounds over $\mathcal{H}$. A prominent one is *Rademacher complexity* relative to the sample $Z\sim\mu^m$

$$\mathfrak{R}(\mathcal{H},Z)=\mathbb{E}_{\sigma\sim\mathcal{U}(\{-1,1\})^m}\Big[\sup_{g\in\mathcal{H}}\frac{1}{m}\sum_{i\in[m]}\sigma_i\phi(X_i)\Big]\qquad Z_i=(X_i,Y_i). \tag{2.78}$$

The supremum on the right will be large if $\mathcal{H}$ contains a function which separates the data points X_i according to the binary labeling $\sigma\in\{-1,1\}^m$. Rademacher complexity thus measures the capacity of $\mathcal{H}$ to separate the data X_i in expectation over random binary labeling. If Rademacher complexity is low, but empirical risk minimization still finds a hypothesis $\phi^{\mathrm{erm}}\in\mathcal{H}$ with low empirical risk, this can be seen as an indication that the hypothesis class is well-aligned with the data distribution at hand. For loss functions taking values in the bounded interval $[0,C]$, [188, Theorem 26.5] shows that with probability at least $1-\delta$ it holds

$$\sup_{\phi\in\mathcal{H}}\mathcal{R}(\phi)-\mathcal{R}_m(\phi)\leq 2\mathfrak{R}(\ell\circ\mathcal{H},Z)+4C\sqrt{\frac{2\log\frac{4}{\delta}}{m}}. \tag{2.79}$$

This approach can deal with infinite hypothesis classes and appears to give clear guidance for learning. If we find a hypothesis class with low Rademacher complexity which contains elements that fit the training data well, (2.79) guarantees generalization. Accordingly, by identifying properties of a model which lead to small Rademacher complexity, we have a principled way of defining effective regularizers. However, with regard to deep learning, [230, 229] have argued that this perspective can not give a complete picture of generalization. They compare the behavior of training convolutional neural networks for image classification between the original dataset and a randomly re-labeled copy of the data. Without any correlation between data and labels, generalization is impossible. However, deep networks trained with stochastic gradient descent (SGD) are still able to achieve vanishing training error, i.e. fit the training data perfectly. Since Rademacher complexity measures the ability of a model class to fit to random data labelings, this

observation renders bounds like (2.79) vacuous in deep learning. Further, [230, 229] have found that (over-)fitting to randomly labeled data does not lead to significant difficulties in the training process. This indicates that reducing the hypothesis class $\mathcal{H}$ in (2.79) to only those models which can be learned via SGD still does not suffice to achieve a non-vacuous bound. Analogous arguments hold for related complexity measures such as Vapnik-Chervonenkis (VC) dimension [210].

2.2.2 PAC-Bayes

The *probably approximately correct Bayesian* construction, *PAC-Bayes* for short, is an alternative way of avoiding the union bound in (2.76). Instead of individual hypotheses $\phi \in \mathcal{H}$, the PAC-Bayesian construction studies the generalization of stochastic classifiers, called *PAC-Bayes posteriors* $\rho \in \mathcal{P}(\mathcal{H})$. In place of a complexity measure for the hypothesis space, like VC dimension or Rademacher complexity, the PAC-Bayesian construction involves the relative entropy $\mathrm{KL}(\rho, \pi)$ between the PAC-Bayes posterior and a reference measure $\pi \in \mathcal{P}(\mathcal{H})$, called *PAC-Bayes prior*. Although ρ will be an update of π, informed by observing additional data, PAC-Bayes prior and posterior are not to be confused with *Bayesian* prior and posterior. In particular, π and ρ are not connected via a likelihood. For brevity, we will sometimes refer to π and ρ as just *prior* and *posterior* in the following. The task of PAC-Bayesian risk certification is to bound the expected risk $\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}(\mathfrak{p})]$ of the posterior. Similar to the above constructions, PAC-Bayesian bounds are built on concentration of measure results like Hoeffding's inequality (Theorem 2.11). Other core ingredients are presented next.

Lemma 2.13 (Markov inequality) *For any nonnegative random variable U and any $a > 0$ it holds*

$$P(U \geq a) \leq \frac{\mathbb{E}[U]}{a}. \tag{2.80}$$

Lemma 2.14 (Chernoff bound) *For any random variable U and $a \in \mathbb{R}$, $t > 0$ it holds*

$$P(U > a) = P(e^{tU} > e^{ta}) \leq e^{-ta}\mathbb{E}[e^{tU}]. \tag{2.81}$$

Lemma 2.15 (Donsker and Varadhan's variational formula [63]) *For any measurable and bounded function $h\colon \mathcal{H} \to \mathbb{R}$ and any distribution $\pi \in \mathcal{P}(\mathcal{H})$, it holds*

$$\log \mathbb{E}_{\mathfrak{p}\sim\pi}[e^{h(\mathfrak{p})}] = \sup_{\rho\in\mathcal{P}(\mathfrak{p})} [\mathbb{E}_{\mathfrak{p}\sim\rho}[h(\mathfrak{p})] - KL(\rho, \pi)] \tag{2.82}$$

and the supremum is attained for the Gibbs measure ρ^* with density

$$\frac{\mathrm{d}\rho^*}{\mathrm{d}\pi}(\mathfrak{p}) = \frac{e^{h(\mathfrak{p})}}{\mathbb{E}_{\psi\sim\pi}[e^{h(\psi)}]}. \tag{2.83}$$

With these preparations, we now recite a PAC-Bayesian theorem of [39] alongside the proof given in [3].

Theorem 2.16 (PAC-Bayesian Bound of [39]) *Suppose a sample $Z = \{(X_i, Y_i)\}_{i\in[m]}$ of m independently drawn data and a PAC-Bayesian prior $\pi \in \mathcal{P}(\mathcal{H})$ independent of the sample are given. For any $\lambda > 0$, $\delta \in (0,1)$ and loss function taking values in $[0, C]$, it holds simultaneously for all PAC-Bayesian posteriors $\rho \in \mathcal{P}(\mathcal{H})$*

$$\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}(\mathfrak{p})] \leq \mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}_m(\mathfrak{p})] + \frac{\lambda C^2}{8m} + \frac{1}{\lambda}\Big(\mathrm{KL}(\rho,\pi) + \log\frac{1}{\delta}\Big) \tag{2.84}$$

with probability at least $1-\delta$ over the draw of the sample.

Proof. For fixed hypothesis $\phi_{\mathfrak{p}} \in \mathcal{H}$, consider the random variables $U_i = \mathbb{E}[\ell(\phi_{\mathfrak{p}}(X_i), Y_i)] - \ell(\phi_{\mathfrak{p}}(X_i), Y_i)$, $i \in [m]$. These variables are viewed as random, by seeing $Z_i = (X_i, Y_i)$ as a random variable following the data distribution. Because draws were independent, the random variables $\{U_i\}_{i\in[m]}$ are also independent and we have

$$\frac{1}{m}\sum_{i\in[m]} U_i = \mathcal{R}(\mathfrak{p}) - \mathcal{R}_m(\mathfrak{p}). \tag{2.85}$$

Since we assumed that the loss function takes values in $[0, C]$, the first variant (2.67) of Hoeffding's inequality gives

$$\mathbb{E}_Z\Big[\exp\Big(\lambda[\mathcal{R}(\mathfrak{p}) - \mathcal{R}_m(\mathfrak{p})]\Big)\Big] \leq \exp\Big(\frac{\lambda^2 C^2}{8m}\Big) \tag{2.86}$$

for any $\lambda > 0$. Taking the expectation of (2.86) with respect to $\mathfrak{p}$ drawn from the PAC-Bayes prior distribution and using Fubini's theorem gives

$$\mathbb{E}_Z\mathbb{E}_{\mathfrak{p}\sim\pi}\Big[\exp\Big(\lambda[\mathcal{R}(\mathfrak{p}) - \mathcal{R}_m(\mathfrak{p})]\Big)\Big] \leq \exp\Big(\frac{\lambda^2 C^2}{8m}\Big). \tag{2.87}$$

We now replace the inner expectation by using Donsker and Varadhan's variational formula (Lemma 2.15)

$$\mathbb{E}_Z\Big[\exp\Big(\sup_{\rho\in\mathcal{P}(\mathcal{H})} \lambda\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}(\mathfrak{p}) - \mathcal{R}_m(\mathfrak{p})] - \mathrm{KL}(\rho,\pi)\Big)\Big] \leq \exp\Big(\frac{\lambda^2 C^2}{8m}\Big). \tag{2.88}$$

Define the random variable

$$\mathcal{U} = \sup_{\rho\in\mathcal{P}(\mathcal{H})} \lambda\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}(\mathfrak{p}) - \mathcal{R}_m(\mathfrak{p})] - \mathrm{KL}(\rho,\pi) - \frac{\lambda^2 C^2}{8m}. \tag{2.89}$$

Then (2.88) implies $\mathbb{E}_Z[\exp(\mathcal{U})] \leq 1$ and Lemma 2.14 consequently gives

$$\mathbb{P}_Z(\mathcal{U} > a) \leq e^{-a}\mathbb{E}_Z[\exp(\mathcal{U})] \leq e^{-a} \tag{2.90}$$

for any $a \in \mathbb{R}$. If we make the choice $e^{-a} = \delta$, (2.90) implies $\mathcal{U} \leq \log\frac{1}{\delta}$ with probability at least $1-\delta$ under the draw of the sample. Expanding the definition of $\mathcal{U}$, the statement $\mathcal{U} \leq \log\frac{1}{\delta}$ becomes

$$\sup_{\rho\in\mathcal{P}(\mathcal{H})} \mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}(\mathfrak{p}) - \mathcal{R}_m(\mathfrak{p})] \leq \frac{\lambda C^2}{8m} + \frac{1}{\lambda}\Big(\mathrm{KL}(\rho,\pi) + \log\frac{1}{\delta}\Big) \tag{2.91}$$

which shows the assertion. □

The bound presented in Theorem 2.16 is not the tightest available PAC-Bayesian result for bounded loss functions and it will not be used in the following chapters. However, we recite this theorem here, because its proof by [3] nicely illustrates PAC-Bayesian techniques without requiring many technical details.

Regarding the numerical evaluation of PAC-Bayesian bounds, key issues are the definition of prior and posterior distributions and the accurate and efficient computation of the *expected* empirical risk $\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}_m(\mathfrak{p})]$, which can be a hard task in practice. We return to these points in Chapter 6.

Part of the appeal of PAC-Bayesian theory from a deep learning perspective is the minimal assumptions made. In this case, the only essential assumptions were boundedness of the loss function, independent draws of the sample from the data distribution and independence of the prior from the data used in the bound. Note that some works in PAC-Bayesian theory weaken these assumptions even further [4, 169, 82, 81]. Even in the comparatively simple case presented here, we did not make any assumptions on the data-generating distribution, the loss function does not need to be differentiable and any posterior is permissible, as long as $\mathrm{KL}(\rho, \pi)$ remains well-defined. Note that this relative entropy between posterior and prior can also be moderate in highly overparameterized settings. The latter is important in the context of deep learning, because with many more parameters than training data, even randomly labeled datasets can be fit exactly using deep classifiers [230, 229] and, contrary to conventional wisdom, these classifiers often do not suffer from severe overfitting in practice [151].

By leveraging the well-positioned PAC-Bayesian theory for the study of deep learning, a milestone has recently been achieved: the first non-vacuous risk bounds for deep classifiers [67, 66, 123]. Subsequent research has also made strides in tightening the available bounds further [161, 162, 21, 47]. A key insight has proven to be the shift from data-free priors to data-dependent ones. While [66] employ a fine-grained analysis using differential privacy to this end, recent works have predominantly opted for the simpler approach of splitting the available data. The first part of the data, which we will refer to as the *training set*, is used to learn π. A separate part, which we call *validation set*, is used to evaluate the bound. In particular, the number m of data in Theorem 2.16 refers to the size of the validation set and expected empirical risk is evaluated on the validation set. Since PAC-Bayesian bounds hold uniformly for all posterior distributions, no matter how they were learned, all data can be used to inform ρ. Since high-probability risk bounds are already sufficient grounds to judge generalization, no separate test data are required in principle.

The shift to focusing on data-dependent priors in PAC-Bayesian methods can be seen as part of a recent development in which classical assumptions and goals of statistical learning theory are reevaluated in light of the unique properties of deep learning. For example, *consistency* is the property of a learning algorithm to converge in probability to a best classifier in the hypothesis space as the sample size approaches infinity. Let $A_m \colon \mathcal{Z}^m \to \mathcal{H}$ denote a learning algorithm which learns a hypothesis in the class $\mathcal{H}$ from a sample of size m. Then A_m is consistent with respect to $\mathcal{H}$ and the data distribution μ if

$$\mathbb{P}_{Z\sim\mu^m}\Big(\mathcal{R}(A_m(Z)) - \inf_{h\in\mathcal{H}} \mathcal{R}(h) > \epsilon\Big) \to 0 \tag{2.92}$$

for every $\epsilon > 0$ as $m \to \infty$ [132]. A classical result in Vapnik-Chervonenkis (VC) theory

states that uniform convergence is not only sufficient, but also *necessary* for consistency of the empirical risk minimization algorithm [210]. However, more recently, [149] have expressed doubt that uniform convergence can be able to explain generalization in deep learning. This discrepancy highlights a change in perspective between classical statistical learning theory and current deep learning practice. The goal of consistency may be too ambitious. Given a dataset drawn i.i.d. from an unknown underlying distribution μ, a deep learning practitioner aims to find a single model which generalizes well – a notably less ambitious goal than finding an algorithm that returns well-generalizing models with high probability if the attempt is repeated. Likewise, the focus needs to be confined to a specific dataset or type of data, rather than on finding an algorithm that universally finds well-generalizing models for any data distribution, which is the topic of *no-free-lunch* theorems [147]. Moreover, since overparameterized deep learning methods can universally achieve essentially zero *training* error, particular focus on *generalization* is required. The way this is frequently pursued in practice involves a laborious process of trial and error, guided by the practioners' intuition, to find an effective combination of network architecture, training regimen and all involved hyperparameters. There is statistical justification for this procedure with regard to the stated goal of finding a *single hypothesis* which generalizes well. The essential statistical tool is holding out a *validation set* of data not used for training. In this case, [104] details how comparing different models based on validation set performance results in a tight, high-probability risk bound for the selected model. This is possible because the variety of trained models whose performance is judged on the validation data is much smaller than the model hypothesis classes specified by all possible realizations of a deep neural network architecture. Consider holding out 10k validation data and, as a simple example, recall the union bound certificate (2.77) for finite hypothesis classes $\mathcal{H}$ with cardinality M

$$\sup_{\phi\in\mathcal{H}} \mathcal{R}(\phi) - \mathcal{R}_m(\phi) \leq C\sqrt{\frac{\log\frac{M}{\delta}}{2m}}. \tag{2.93}$$

Suppose the network architecture, learning rate, dropout rate and weight decay need to be tuned. If we test 10 values for each hyperparameter, a grid search compares 10^4 models. For a validation set of size $m = 10^4$ and 0/1 loss bounded by $C = 1$, the bound (2.93) reads

$$\sup_{\phi\in\mathcal{H}} \mathcal{R}(\phi) - \mathcal{R}_m(\phi) \leq 0.0263 \tag{2.94}$$

with probability at least $1 - \delta = 0.99$, a convincingly tight bound. Moreover, once a single hypothesis is selected, one still has access to additional held-out test data. The single-hypothesis analog to (2.93) was presented in (2.72). Assuming 10k additional test data, we find

$$\mathcal{R}(\phi^*) - \mathcal{R}_{m_{\text{test}}}(\phi^*) \leq 0.0152 \tag{2.95}$$

with probability at least $1 - \delta = 0.99$ for the single hypothesis $\phi^* \in \mathcal{H}$ selected through hyperparameter tuning. This validation method allows for arbitrary network architectures and training procedures, including highly overparameterized models trained with variants of stochastic gradient descent.

Although bounds like (2.93) or the (more general) ones presented in [104] provide some statistical justification for the practice of hyperparameter tuning, they merely examine models after training and do not provide theoretical insight to *inform the training procedure itself*. In contrast, PAC-Bayesian bounds hold uniformly for all posterior distributions over the hypothesis class, allowing the use of the *risk certificate as a training objective* for the posterior. Thus, PAC-Bayesian bounds can inform the training procedure and provide effective regularization to achieve better generalization. However, the tightness of risk bounds and the selection of a prior are crucial to this end.

3 Assignment Flows

In this chapter, we introduce the *assignment flow* approach to data labeling on graphs. It consistutes a geometric mathematical framework with broad applicability to structured prediction problems in the scope of this thesis. Assignment flows were first proposed by [11]. For general reference and overview, refer to [185]. Here, we present a broad perspective which starts with the most general version of assignment flows and is later specialized for specific problem instances. We start with Section 3.1 on graphical models and Section 3.2 on replicator dynamics. These sections provide context and motivation for the assignment flow approach introduced in Section 3.3.

3.1 Graphical Models for Structured Prediction

Modelling complex joint distributions of many interacting variables has long been a core problem in statistical physics. A prototypical example is the *Ising model* of ferromagnetism [100]. Suppose a lattice of n microscopic ferromagnets, each either in state *up* or in state *down*. At lattice site $i \in [n]$, a *unary* energy $\theta_j^{(i)} \geq 0$ is associated with the ferromagnet being in state $j \in [c]$, $c = 2$. In addition, along each lattice edge $ik \in \mathcal{E} \subseteq [n] \times [n]$, the combination of states $(j, l) \in [c]^2$ at connected lattice sites is associated with a *pairwise* interaction energy $\theta_{jl}^{(ik)}$, which in the Ising model is zero for matching states and a positive constant $\theta^p > 0$ for differing states

$$\theta_{jl}^{(ik)} = \begin{cases} 0, & \text{if } j = l, \\ \theta^p, & \text{else.} \end{cases} \tag{3.1}$$

This local interaction along edges compounds to a complex combinatorial structure of total system energy. If the system is at thermal equilibrium and has temperature $\lambda > 0$, the

probability of finding it in state $\alpha \in \{0,1\}^n$ is governed by the *Gibbs distribution*[1] [116]

$$p_\alpha = \frac{1}{Z(\theta)} \exp\Big(-\frac{1}{\lambda} E_\theta(\alpha)\Big), \qquad \theta = (\theta^{(1)}, \dots, \theta^{(n)}, \theta^p) \tag{3.2a}$$

$$E_\theta(\alpha) = \sum_{i\in[n]} \theta^{(i)}_{\alpha_i} + \sum_{ik\in\mathcal{E}} \theta^{(ik)}_{\alpha_i\alpha_k} \tag{3.2b}$$

$$Z(\theta) = \sum_{\alpha\in[c]^n} \exp\Big(-\frac{1}{\lambda} E_\theta(\alpha)\Big). \tag{3.2c}$$

Even if we have perfect knowledge of system energy for every configuration $\alpha \in [c]^n$, the normalizing constant (3.2c), called *partition function* is still an intractable quantity in general, because it is composed of a combinatorial number of summands. Within the framework provided by probabilistic models like (3.2), the following three tasks are commonly considered.

MAP Inference Natural point estimates of system state are the modes of the distribution (3.2). They are found as configurations with minimal energy, called maximum a posteriori (MAP) estimates. Comparing relative energy between system configurations does not require knowledge of the partition function. However, MAP estimation is still a hard problem, because a combinatorial number of configurations need to be compared. In fact, outside of special problem subclasses, exact solution is NP-hard [30] and approximation of the solution is exp-APX complete [124].

Probabilistic Inference Beyond point estimates, marginals and other statistics of (3.2), as well as the ability to draw samples are of interest. These tasks are naturally linked to quantifying uncertainty. For instance, a physical system at high temperature $\lambda \gg 0$ typically has a large number of configurations with similar probability. It is thus very uncertain in which state one will observe the system and point estimates are unreliable, even if all modes are known. In contrast, low temperature $0 < \lambda \ll 1$ leads to large discrepancies between the probability of low-energy versus high-energy configurations. It is thus more likely that the system is found in one of the low-energy states, making point estimates of low-energy configurations a reliable indicator.

Learning If we do not have knowledge of system energy for each configuration, but we instead have access to a set of samples from the distribution (3.2), we may make a parametric ansatz for the energy function like (3.2b) and learn parameters to match the distribution of samples. Learning is generally a more difficult task than inference. In fact, repeated inference is usually a part of learning procedures [62, 204].

All three tasks described above are hard combinatorial problems in general. Linear programming relaxations can be employed for MAP estimation. Instead of looking for an

[1] If the term *energy* is meant in a physical sense, the exponent in (3.2a) needs to be scaled by Boltzmann's constant, which we omit for simplicity of notation.

energy-minimizing configuration in the discrete set $[c]^n$, soft assignment of classes $[c]$ to each of the n variables are encoded as vectors

$$W_i = (W_{i1}, \dots, W_{ic}) \in \mathbb{R}^c_{\geq 0},\ \langle W_i, \mathbb{1}_c \rangle = 1, \qquad i \in [n] \tag{3.3}$$

subject to probability simplex constraints. By additionally associating each edge of the underlying graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with pairwise probability distributions

$$W^{ik} \in \mathbb{R}^{c \times c}_{\geq 0},\ \langle W^{ik}, \mathbb{1}_c \mathbb{1}_c^\top \rangle = 1, \qquad ik \in \mathcal{E} \tag{3.4}$$

a relaxation of the MAP inference problem is found as the linear program

$$\min_{(\{W_i\}_{i\in[n]}, \{W^{ik}\}_{ik\in\mathcal{E}}) \in \mathcal{L}_{\text{loc}}} \sum_{i\in[n]} \langle W_i, \theta^{(i)} \rangle + \sum_{ik\in\mathcal{E}} \langle W^{ik}, \theta^{(ik)} \rangle \tag{3.5}$$

where $\mathcal{L}_{\text{loc}}$ is called the *local polytope* which is the subset of vectors in $(\mathbb{R}^c)^n \times (\mathbb{R}^{c\times c})^{|\mathcal{E}|}$ satisfying the respective constraints of (3.3) and (3.4) as well as the marginalization constraints

$$W^{ik}\mathbb{1}_c = W_i, \quad (W^{ik})^\top \mathbb{1}_c = W_k, \qquad ik \in \mathcal{E}. \tag{3.6}$$

The linear program (3.5) is a relaxation of MAP inference, because its feasible set is enlarged by dropping the integrality constraints

$$W_i \in \{0,1\}^c, \qquad i \in [n]. \tag{3.7}$$

Consequently, even though a solution to (3.5) can be computed in polynomial time [122], this solution will generally not satisfy (3.7). Therefore, *rounding* to the nearest integer solution is required to find a discrete system configuration and it can be shown that the result of this procedure can be arbitrarily far from the MAP estimate [184, Example 4.5]. Since learning typically involves repeated inference, it is plagued by the same inaccurate approximation.

Assignment flows build on the perspective of graphical models for structured prediction problems. In Section 3.3, we will return to the problems of *inference* and *learning* posed above and discuss how assignment flows can work around the computational difficulties associated with these tasks. We start with a short interlude on *population dynamics*, which is a topic typically discussed in the language of *game theory*.

3.2 Replicator Dynamics

Suppose a large population of players, each using one of $c > 0$ available strategies. Define a game as two-player interactions leading to payoff for each player. Let $x > 0$ denote the number of players in the population and let x_j denote the number of players employing strategy $j \in [c]$. Assume that x_j evolves over time proportional to the expected payoff f_j of strategy j when playing against a random player in the population

$$\dot{x}_j = x_j f_j. \tag{3.8}$$

This assumption was put forward by [199] along with biological motivation. If payoff is associated with Darwinian fitness, then individuals employing a successful strategy will replicate at a high rate and their offspring will again be likely to employ the same strategy. We can now study the evolution of relative frequencies $p_j = \frac{x_j}{x}$ in the population

$$\dot{p}_j = \frac{\dot{x}_j}{x} - \frac{\dot{x}}{x}\frac{x_j}{x} = p_j f_j - \frac{\dot{x}}{x} p_j = p_j f_j - \frac{p_j}{x} \sum_{l \in [c]} x_l f_l = p_j \Big(f_j - \sum_{l \in [c]} p_l f_l \Big) \tag{3.9}$$

for each strategy j. Suppose the average payoff $f \colon \mathcal{S}_c \to \mathbb{R}^c$ can be written as a function of the relative frequencies $p \in \mathcal{S}_c$. Vectorization of (3.9) reads

$$\dot{p} = \operatorname{Diag}(p) f(p) - \langle p, f(p) \rangle p = R_p[f(p)] \tag{3.10}$$

which is called the *replicator equation* due to its roots in biological replication mentioned above.

As a simple example, assume that the payoff results from two-player interactions and only depends on each players own strategy $j \in [c]$ and the strategy $l \in [c]$ employed by their opponent. The game is now characterized by the *payoff matrix* $B \in \mathbb{R}^{c \times c}$ which lists the payoffs for all c^2 combinations of strategies in two-player interaction as entries B_{jl}. The average payoff of a player with strategy j playing against a random player in the population reads

$$f_j = \sum_{l \in [c]} \frac{x_l}{x} B_{jl} = \sum_{l \in [c]} B_{jl} p_l. \tag{3.11}$$

Note that payoff in (3.11) is indeed a function of p, no knowledge of x_j or x is required to evaluate f_j. The resulting *linear fitness* replicator dynamics read

$$\dot{p} = R_p[Bp]. \tag{3.12}$$

Dynamical systems with the shape (3.10) evolve a state which represents relative frequencies of strategies in a population. This state naturally lives in the simplex of discrete probability vectors

$$\Delta_c = \{ p \in \mathbb{R}^c_{\geq 0} \colon \langle p, \mathbb{1}_c \rangle = 1 \} \tag{3.13}$$

which we already saw as relaxation domain of each discrete variable in (3.3). Its relative interior

$$\mathcal{S}_c = \{ p \in \mathbb{R}^c_{> 0} \colon \langle p, \mathbb{1}_c \rangle = 1 \} \tag{3.14}$$

can be regarded as a Riemannian manifold with the Fisher-Rao metric (recall Section 2.1). This perspective is natural in the context of replicator equations for the following reason. If players are anonymous, the payoff of two-player interactions does not depend on the order of players and the payoff matrix B is symmetric. As a result, the payoff Bp can be written as gradient of the potential

$$J(p) = \frac{1}{2} \langle p, Bp \rangle \tag{3.15}$$

which itself has a nice interpretation as (half) the mean payoff achieved by players in the population. If (3.14) is seen as Riemannian manifold equipped with the Fisher-Rao metric, then the replicator dynamics (3.10) precisely generate the Riemannian gradient ascent flow of (3.15).

Theorem 3.1 (Proposition 1 of [11]) *Let $J\colon \mathbb{R}^c \to \mathbb{R}$ be a smooth function and $\widetilde{J}\colon \mathcal{S}_c \to \mathbb{R}$ its restriction to $\mathcal{S}_c \subseteq \mathbb{R}^c$. Then the Riemannian gradient of $\widetilde{J}$ at $p \in \mathcal{S}_c$ is given by*

$$\nabla_p \widetilde{J}(p) = R_p[\nabla J(p)] \tag{3.16}$$

where ∇J denotes the (Euclidean) gradient of J.

3.3 Assignment Flows

3.3.1 Dynamical Systems on the Assignment Manifold

Equipped with background on graphical models and replicator dynamics, we return to the structured prediction setting of n coupled discrete random variables, each able to assume one of c class values. Without restriction of generality, we can associate the interdependencies of random variables in their joint distribution with a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Each node represents one variable ($|\mathcal{V}| = n$) and an edge between two nodes represents a dependency to be specified. For instance, a Markov random field is characterized by conditional independence of random variables, given values for a separating set of variables in $\mathcal{G}$. The association of a graph in itself does not restrict generality. The conditional independence structure already present in the joint distribution of a Markov random field is merely encoded into its adjacency relation on $\mathcal{G}$. If all variables are independent, the edge set $\mathcal{E}$ is empty. At the other extreme, if no conditional independence exists between any set of variables, the graph $\mathcal{G}$ is fully connected.

A particular class of structured prediction problems can be formalized as *data labeling* on graphs. Given some data on each graph node, the data labeling task is to infer one of c classes on each node. More formally, let $\mathcal{F}$ be a metric space which acts as domain for data and let

$$\mu^{\mathrm{LD}} \in \mathcal{P}(\mathcal{F}^n \times [c]^n) \tag{3.17}$$

be a joint distribution of labeled data on $\mathcal{G}$. Data labeling is the task of learning the conditional distributions

$$\mu(\cdot|x) \in \mathcal{P}([c]^n) \tag{3.18}$$

given data $x \in \mathcal{F}^n$. A comparatively simple case is deterministic data labeling which is not subject to uncertainty. In this case, each data vector x is deterministically linked to a labeling $\alpha \in [c]^n$ and the conditional distributions (3.18) are discrete Dirac distributions δ_α, $\alpha \in [c]^n$. This means all probability mass is concentrated on a single labeling α, but α still depends on the data x. Generally, distributions in $\mathcal{P}([c]^n)$ are high-dimensional, combinatorial objects. To represent $p \in \mathcal{P}([c]^n)$ as a probability vector, one needs c^n entries, which is typically intractable. For this reason, a core problem of structured prediction is to parameterize low-dimensional subsets of $\mathcal{P}([c]^n)$. We return to this fundamental question in Chapters 4 and 5. The case of deterministically linking data vectors to labelings is comparatively simple, because discrete Dirac distributions have a simple low-dimensional representation as n class indices, each in the range $[c]$. This is still highly non-trivial and useful in many applications. As an example, consider dense image segmentation. Let

each image pixel be associated with a node of $\mathcal{G}$ and suppose there are c segments to be found. The data space $\mathcal{F}$ can be identified with some color space like the cube $[0,1]^3$ of RGB values and a whole image corresponds to a data vector $x \in \mathcal{F}^n$. The assumption of conditional distributions (3.18) being discrete Dirac measures then amounts to the assumption that each image has a deterministic segmentation. This may not be the case in situations where the data are ambiguous and multiple segmentations are possible for one image, but it is still a relatively weak assumption in practice.

Because discrete Dirac measures can be identified with labelings of all nodes, each with a class in $[c]$, the deterministic data labeling scenario is amenable to relaxation akin to (3.5). We will keep the interaction along graph edges unspecified for now and only consider the relaxed domain of probability vectors (3.3) which encode soft class assignment separately on each node

$$\Delta_c \times \cdots \times \Delta_c, \qquad (n \text{ factors}). \tag{3.19}$$

The set (3.19) can be regarded as a polytope embedded in $\mathbb{R}^{nc}$ generated as convex hull of extremal points

$$(e_{\alpha_1}, \ldots, e_{\alpha_n}), \qquad \alpha \in [c]^n. \tag{3.20}$$

Thus, we have identified a one-to-one correspondence between discrete Dirac distributions, labelings of data and extremal points of (3.19)

$$\delta_\alpha \equiv \alpha \equiv (e_{\alpha_1}, \ldots, e_{\alpha_n}), \qquad \alpha \in [c]^n. \tag{3.21}$$

The core idea of assignment flows is to represent inference of discrete Dirac distributions solving data labeling (3.18) as a gradual process which evolves states on (3.19) toward extremal points (3.20). This in turn achieves the stated goal of infering discrete Dirac distributions due to the correspondence (3.21). The guiding design priciple for such inference dynamics will be to build on replicator equations (3.10). This choice has strong motivation from multiple angles. First, replicator dynamics have been a topic of intense research for many years in game theory. Thus, interpretation in game theoretical or biological terms will provide a natural avenue for better understanding of the constructed systems. For example, in Chapter 4, we will clarify how the game-theoretical notions of evolutionary stability and Nash equilibrium fit into the data-labeling picture. Second, replicator dynamics have a natural shape relative to the information geometry of their underlying domain as is apparent from the fact that they generate Riemannian ascent flows (3.15). The Fisher information metric is the Hessian metric [5] of negative entropy

$$-\nabla^2 H(p) = \nabla^2 \langle p, \log p \rangle = \operatorname{Diag}\left(\frac{\mathbb{1}_c}{p}\right) \tag{3.22}$$

and since

$$\nabla_p^2 \operatorname{KL}(p, p_0) = \nabla^2 \Big(- H(p) - \langle p, \log p_0 \rangle \Big) = \operatorname{Diag}\left(\frac{\mathbb{1}_c}{p}\right), \tag{3.23}$$

this makes it a measure for how the information content changes between nearby distributions. Thus, replicator dynamics modify gradient fields derived from an intuition about (Euclidean) probability vectors in a way which is maximally meaningful to change in the information content of the evolved probabilistic state.

In Section 3.2, we have seen that the state space of replicator dynamics is naturally seen as a Riemannian manifold $\mathcal{S}_c$ equipped with Fisher-Rao geometry. Since we are now interested in the case of *multiple* coupled discrete variables, we define the *assignment manifold*

$$\mathcal{W} = \mathcal{S}_c \times \cdots \times \mathcal{S}_c, \qquad (n \text{ factors}) \tag{3.24}$$

as the product manifold of n probability simplices $\mathcal{S}_c$. We call elements of $\mathcal{W}$ *assignment matrices* due to their representation as row-stochastic matrices in the ambient space $\mathbb{R}^{n\times c}$. The *barycenter* of $\mathcal{W}$ is denoted as

$$\mathbb{1}_{\mathcal{W}} = \frac{1}{c}\mathbb{1}_n\mathbb{1}_c^\top \in \mathcal{W} \tag{3.25}$$

and the tangent space at $\mathbb{1}_{\mathcal{W}}$ is identified with the vector space

$$T_0\mathcal{W} = \{V \in \mathbb{R}^{n\times c} \colon V\mathbb{1}_c = 0\} = (T_0\mathcal{S}_c)^n. \tag{3.26}$$

$\mathcal{W}$ has trivial tangent bundle

$$T\mathcal{W} \equiv \mathcal{W} \times T_0\mathcal{W} \tag{3.27}$$

and the *product* Fisher-Rao metric on $T_0\mathcal{W}$ at $W \in \mathcal{W}$ reads

$$g_W(V,U) = \langle V, U\rangle_W = \Big\langle \frac{V}{W}, U \Big\rangle. \tag{3.28}$$

For $W \in \mathcal{W}$, let $\mathcal{R}_W$ denote the operator which applies the replicator R_{W_i} separately on each node $i \in [n]$. Similarly, let

$$\exp_W(V), \qquad V \in T_0\mathcal{W} \tag{3.29}$$

denote the map which applies (2.45) separately on each node and re-use the symbol $\Pi_0 U$ to denote the projection of $U \in \mathbb{R}^{n\times c}$ to $T_0\mathcal{W}$. Analogous to (2.51), we have

$$\exp_W \circ \Pi_0 = \exp_W \tag{3.30}$$

which allows to extend the domain of $\exp_W$ to $\mathbb{R}^{n\times c}$. Similarly, one easily verifies that

$$\mathcal{R}_W \circ \Pi_0 = \Pi_0 \circ \mathcal{R}_W = \mathcal{R}_W \tag{3.31}$$

which allows to extend the domain of $\mathcal{R}_W$ to $\mathbb{R}^{n\times c}$. We will still refer to these objects as *lifting map* and *replicator operator*, respectively. In order to specify interaction along graph edges, we now define

$$F \colon \mathcal{W} \to \mathbb{R}^{n\times c} \tag{3.32}$$

called *payoff* or *fitness* function, such that $F(W)_i$ depends on W_j exactly if $ij \in \mathcal{E}$. This finally specifies the dynamical system

$$\dot{W}(t) = \mathcal{R}_{W(t)}\Big[F\Big(W(t)\Big)\Big], \qquad W(0) = W_0, \tag{3.33}$$

on $\mathcal{W}$, whose solution is called *assignment flow*. Note, that assignment flow dynamics (3.33) for data labeling are the natural result of the following design paradigms.

1. **Gradual decision-making**. The complexity of inference is broken down over time t by modelling the generator F of a dynamical process which gradually arrives at a decision.
2. **Probabilistic state**. At any given time t, the state of decision-making is a soft assignment of classes to nodes, i.e. a probability vector for each node.
3. **Information geometry**. Change in the probabilistic assignment state is viewed as change in information content represented by it.

In particular, (3.33) primarily specifies the language in which inference algorithms are written, it does not yet express problem-specific considerations and it only restricts generality by fixing the dimension of the underlying state.

Inkeeping with the goal of infering discrete Dirac distributions, payoff functions need to be chosen such that (3.33) drives assignment state to extremal points of $\mathcal{W}$. Convergence and stability of assignment flows were studied by [227]. A central result is the following.

Theorem 3.2 (Theorem 2 of [227]) *Let $\Omega \in \mathbb{R}^{n\times n}$, $\Omega = \Omega^\top$ have non-negative entries and positive diagonal entries. Then the assignment flow generated by* (3.33) *with linear payoff function $F(W) = \Omega W$ converges to an extremal point of $\mathcal{W}$ for almost all initializations W_0.*

In practice, convergence of assignment flows to extremal points is observed for many more general payoff functions. We return to this topic under weaker assumptions in the game-theoretical framework of Chapter 4. With regard to stability, the same class of assignment flows also admits a bound on the Lipschitz constant of its flow map. This is elaborated in Appendix C.1.

The following sections return to the tasks of *inference* and *learning* posed in the context of graphical models in Section 3.1.

3.3.2 Inference by Numerical Integration

Once a payoff function $F\colon \mathcal{W} \to \mathbb{R}^{n\times c}$ and initialization W_0 is specified such that (3.33) converges to an extremal point of $\mathcal{W}$, inference amounts to numerical integration of (3.33). In the curved geometry of $\mathcal{S}$, this is best treated using specialized methods [225] in order to ensure that each numerical iterate satisfies the constraints of $\mathcal{W}$. The key will be to parameterize (3.33) on the tangent space $T_0\mathcal{W}$ by using the lifting map. This was done in [225, Proposition 3.1] which we recite as the following theorem.

Theorem 3.3 (Tangent Space Parameterization) *For any initialization W_0 and payoff function $F\colon \mathcal{W} \to \mathbb{R}^{n\times c}$, the solution to the initial value problem* (3.33) *admits the representation $W(t) = \exp_{W_0}(V(t))$, $t \geq 0$ where $V(t) \in T_0\mathcal{W}$ solves*

$$\dot{V}(t) = \Pi_0 F(\exp_{W_0}(V(t))), \qquad V(0) = 0 \in T_0\mathcal{W}. \tag{3.34}$$

Note that due to the lifting map action property (2.48), the parameterization (3.34) can equivalently be done at the barycenter

$$\dot{V}(t) = \Pi_0 F(\exp_{\mathbb{1}_\mathcal{W}}(V(t))), \qquad V(0) = \exp^{-1}_{\mathbb{1}_\mathcal{W}}(W_0). \tag{3.35}$$

Numerical integration of (3.33) can thus be performed by integrating (3.34) or (3.35) and lifting the result to the assignment manifold $\mathcal{W}$. Since the latter are ordinary differential equations in the flat and unbounded vector space $T_0\mathcal{W}$, standard methods are applicable. For example, the explicit Euler method for integrating (3.34) with step-length $h > 0$ reads

$$V(t+h) = V(t) + h\Pi_0 F(\exp_{W_0}(V(t))) \tag{3.36}$$

which corresponds to the *geometric* Euler scheme

$$\begin{aligned} W(t+h) &= \exp_{W_0}(V(t+h)) && \text{(3.37a)}\\ &= \exp_{W_0}(V(t) + h\Pi_0 F(\exp_{W_0}(V(t)))) && \text{(3.37b)}\\ &= \exp_{\exp_{W_0}(V(t))}(h\Pi_0 F(\exp_{W_0}(V(t)))) && \text{(3.37c)}\\ &= \exp_{W(t)}(h\Pi_0 F(W(t))). && \text{(3.37d)} \end{aligned}$$

This also illustrates the equivalence of (3.34) and (3.35) because for $W(t) = \exp_{\mathbb{1}_\mathcal{W}}(V(t))$,

$$\begin{aligned} W(t+h) &= \exp_{\mathbb{1}_\mathcal{W}}(V(t+h)) && \text{(3.38a)}\\ &= \exp_{\mathbb{1}_\mathcal{W}}(V(t) + h\Pi_0 F(\exp_{\mathbb{1}_\mathcal{W}}(V(t)))) && \text{(3.38b)}\\ &= \exp_{\exp_{\mathbb{1}_\mathcal{W}}(V(t))}(h\Pi_0 F(\exp_{\mathbb{1}_\mathcal{W}}(V(t)))) && \text{(3.38c)}\\ &= \exp_{W(t)}(h\Pi_0 F(W(t))) && \text{(3.38d)} \end{aligned}$$

generates the same geometric Euler iterates on $\mathcal{W}$. The same equivalence also holds for other numerical integrators, see Appendix C.2 for details.

Under suitable conditions, the result of integrating (3.33) can be rounded to an extremal point of $\mathcal{W}$ after finite time [227]. This inference process identifies a discrete Dirac measure through the correspondence (3.21). If the desired conditional distribution (3.18) of labelings given data is not a discrete Dirac measure, an extension of the assignment flow framework is required. We turn to this more general problem of probabilistic inference in Chapter 5. We conclude that, in contrast to graphical models, the (point estimate) inference process defined by assignment flows is numerically tractable with arbitrary precision if payoff functions can be evaluated efficiently and convergence to extremal points occurs for the given initialization W_0. Under these conditions, the assignment flow approach simultaneously realizes two aspects of graphical model inference in a single smooth process. First, coupling between random variables is facilitated through interaction along graph edges. Second, the dynamical system (3.33) drives soft assignment states to hard class assignments over time, aided by the underlying Fisher-Rao geometry, which prevents many vector fields from generating flows with stable limits at non-extremal boundary points of Δ_c^n.

3.3.3 Learning Payoff Functions

A further numerical advantage of the assignment flow approach is its inherent smoothness, which allows to learn payoff functions from data. This was studied by [98] through the lense of adjoint sensitivity and by [226] via linearization. Here, we briefly introduce the

first approach, which is more relevant to the following chapters than the second. As a starting point, consider $m > 0$ independently drawn samples

$$(x_k, y_k) \sim \mu^{\mathrm{LD}}, \qquad k \in [m] \tag{3.39}$$

each consisting of a data vector $x_k \in \mathcal{F}$ and a labeling $y_k \in [c]^n$. Assume again that all conditional distributions (3.18) of labelings given data are discrete Dirac measures. This ensures that there exists a deterministic relationship between data and labelings to be learned. Further, suppose we have a way to associate data vectors in $\mathcal{F}$ with initial assignment states $W_0 \in \mathcal{W}$. Such an association will be discussed in Section 3.3.4. We now define some set $\mathcal{H}$ of functions $F_{\mathfrak{p}} \colon \mathcal{W} \to \mathbb{R}^{n \times c}$, each uniquely identified by parameters $\mathfrak{p}$ in a Euclidean parameter space and some loss function $\mathcal{L} \colon \mathcal{W} \times [c]^n \to \mathbb{R}$ which measures the deviation of soft assignments from labelings. *Learning* a payoff function can then be written as minimizing the risk

$$\mathbb{E}_{(x,y)\sim\mu^{\mathrm{LD}}}\Big[\mathcal{L}(\psi(W_0(x);\mathfrak{p}), y)\Big] \tag{3.40}$$

where $\psi(W_0;\mathfrak{p}) = \psi_T(W_0;\mathfrak{p})$ refers to the flow map $\psi_T \colon \mathcal{W} \to \mathcal{W}$ of (3.33) for a fixed time horizon $T > 0$, i.e.

$$\psi_T(W_0;\mathfrak{p}) = W(T) \tag{3.41a}$$

$$\text{s.t. } \dot{W}(t) = \mathcal{R}_{W(t)}[F_{\mathfrak{p}}(W(t))] \quad \forall t \in [0,T], \quad W(0) = W_0. \tag{3.41b}$$

Since we only have access to μ^{LD} through samples, we consider the empirical risk counterpart of (3.40)

$$\frac{1}{m} \sum_{k\in[m]} \Big[\mathcal{L}(\psi(W_0(x_k);\mathfrak{p}), y_k)\Big]. \tag{3.42}$$

Efficient minimization of the empirical risk (3.42) can be approached through stochastic gradient methods. To this end, we need to compute the gradient of loss with respect to parameters. Since evaluation of the loss involves integration of (3.41b) forward in time, the gradient can in principle be approximated by differentiating a numerical integration scheme with respect to the parameters. We will call this approach *discretize-then-optimize*, because discretization of the ODE (3.41b) proceeds differentiation of the discretized system for the purpose of optimizing parameters. This approach is easy to implement by using automatic differentiation software [16]. However, it entails a large memory footprint in practical applications because the system state $v(t_i)$ needs to be saved for all discretization points, which limits scalability. The problem of computing the gradient in question is well-known in the context of parameter estimation and optimal control [203] and specialized methods have been developed to avoid *discretize-then-optimize*. For simpler presentation, we will consider the tangent space parameterization (3.34) which evolves on the flat and unbounded vector space $T_0\mathcal{W}$. For fixed initialization $W_0 \in \mathcal{W}$, define

$$\widetilde{\mathcal{L}} \colon T_0\mathcal{W} \times [c]^n \to \mathbb{R}, \tag{3.43a}$$

$$(v, y) \mapsto \widetilde{\mathcal{L}}(v, y) = \mathcal{L}(\exp_{W_0}(v), y) \tag{3.43b}$$

and the tangent flow map

$$\widetilde{\psi}_T(W_0;\mathfrak{p}) = V(T) \tag{3.44a}$$
$$\text{s.t. } \dot{V}(t) = \Pi_0 F_{\mathfrak{p}}(\exp_{W_0}(V(t)))] \quad \forall t \in [0,T], \quad V(0) = 0. \tag{3.44b}$$

Then minimization of the tangent empirical risk

$$\frac{1}{m}\sum_{k\in[m]} \widetilde{\mathcal{L}}(\widetilde{\psi}(W_0(x_k);\mathfrak{p}), y_k) \tag{3.45}$$

is equivalent to minimizing (3.42) by Theorem 3.3. Define the shorthand notation

$$f(V(t),\mathfrak{p}) = \Pi_0 F_{\mathfrak{p}}(\exp_{W_0}(V(t)))]. \tag{3.46}$$

The following theorem, which has its origin in the theory of parameter estimation and optimal control, provides an alternative way of computing the desired parameter gradient.

Theorem 3.4 (Theorem 6 of [98]) *The gradient of* (3.45) *is given by*

$$\frac{1}{m}\sum_{k\in[m]} \partial_{\mathfrak{p}}\widetilde{\mathcal{L}}(\widetilde{\psi}(W_0(x_k);\mathfrak{p})) = \frac{1}{m}\sum_{k\in[m]} \int_0^T \mathrm{d}_{\mathfrak{p}} f(V(t),\mathfrak{p})^\top \lambda(t)\mathrm{d}t \tag{3.47}$$

where $\mathrm{d}_{\mathfrak{p}} f$ *denotes the differential of* f *with respect to* $\mathfrak{p}$ *and* $V(t)$, $\lambda(t)$ *solve the* adjoint *differential equation*

$$\dot{V}(t) = f(V(t),\mathfrak{p}), \qquad V(0) = 0, \tag{3.48a}$$
$$\dot{\lambda}(t) = -\mathrm{d}_V f(V(t),\mathfrak{p})^\top \lambda(t), \qquad \lambda(T) = \partial_V \widetilde{\mathcal{L}}(V)|_{V=\widetilde{\psi}(W_0(x_k);\mathfrak{p})}. \tag{3.48b}$$

By choosing a quadrature for the integral (3.47), Theorem 3.4 allows to compute the desired gradient without the need to save system state at all discretization points. This constitutes an *optimize-then-discretize* approach, because differentiation in continuous-time proceeds discretization. In general, the resulting gradient approximation is different from *discretize-then-optimize*. However, it has been shown [98, 179] that for particular *symplectic* integrators, discretization commutes with optimization, i.e. both orders of operation yield the same gradient.

An image labeling example is shown in Figure 3.1. Here, the data is modeled by an assignment flow with shape to be defined in (3.63) and graph adjacency representing local pixel neighborhoods (3×3). Starting from a noisy, high-entropy assignment of pixels to color prototypes ($c = 47$), the goal is to learn parameters $\mathfrak{p}$ such that the state is driven to a given noise-free assignment after the fixed integration time $T = 15$. In this example, parameters are entries of a label interaction matrix $B \in \mathbb{R}^{c\times c}$. We initialized B as identity matrix and performed 100 steps of the Adam optimizer to minimize cross-entropy between the ground truth assignment and the final state reached by the assignment flow. This training procedure is memory efficient – for the 256×256 pixel image in Figure 3.1,

[2]The original artwork used in Figure 3.1 was designed by dgim-studio / Freepik.

Figure 3.1: *Left*: Noisy input assignment of $c = 47$ colors to the pixels of an image. *Center*: Limit of an EGN flow (3.63) with *learned* interaction in 3×3 pixel neighborhoods. *Right*: Ground truth noise-free color assignment.[2]

training takes less than a minute on a laptop computer and requires around 1.3GB of vRAM.

In deep learning, parameterized dynamical systems like the ones defined by parameterized payoff functions $F_{\mathfrak{p}} \colon \mathcal{W} \to \mathbb{R}^{n \times c}$ are known as *neural ordinary differential equations* (nODE) [43]. If the vector field defining these systems satisfies a Lipschitz condition, the classic theorem of Picard-Lindelöf implies that integral curves exist uniquely. As a result, every point of the domain belongs to a unique integral curve and hence, the flow map for fixed time $T > 0$ is invertible. This is the starting point for the construction of continuous-time measure transport methods of representing complex data distributions, called *continuous normalizing flows* (CNF). We return to this topic in Chapter 5.

3.3.4 Examples of Assignment Flows

W-flow The first assignment flow which was contextualized as such is the image labeling method proposed by [11]. The authors assume that graph nodes are linked to pixels of an image. Feature vectors

$$f_i \in \mathcal{F}, \qquad i \in [n] \tag{3.49}$$

in a metric feature space $\mathcal{F}$ are given for each node. These feature vectors are assumed to be directly informative for the labeling problem at hand and a metric $d_{\mathcal{F}}$ on $\mathcal{F}$ is known which preserves the informative nature of features. For low-level image processing tasks, these requirements may be satisfied by largely unprocessed data such as color values. For higher-level semantic tasks, this likely requires preprocessing raw data to extract informative features. For each class, a prototypical feature vector

$$f_j^* \in \mathcal{F}, \qquad j \in [c] \tag{3.50}$$

is selected which ultimately allows to abstract from the feature space $\mathcal{F}$. To this end, define the *distance matrix* $D \in \mathbb{R}^{n \times c}$ which collects pairwise distances

$$D_{ij} = d_{\mathcal{F}}(f_i, f_j^*), \qquad (i, j) \in [n] \times [c] \tag{3.51}$$

between node features and prototypes (3.50). Then the data is abstracted from the feature space $\mathcal{F}$ by lifting distance vectors at the current assignment state

$$L(W) = \exp_W(-D). \tag{3.52}$$

Note the implicit projection of $-D$ to the tangent space $T_0\mathcal{W}$ in accordance with (3.30). The states (3.52) are subsequently averaged in each graph neigborhood, defining the *similarity* map

$$S_i(L) = \mathrm{mean}_\Omega\{L_j \colon j \in \mathcal{N}_i\}, \qquad i \in [n]. \tag{3.53}$$

Here, the mean is weighted by entries of the row-stochastic adjacency matrix Ω. It can be taken with respect to the Levi-civita connection on $\mathcal{W}$, or alternately with respect to the e-connection, which is computationally more efficient. The resulting assignment flow dynamics, called *W-flow*, read

$$\dot{W}(t) = \mathcal{R}_{W(t)}[S(L(W(t)))], \qquad W(0) = \mathbb{1}_{\mathcal{W}}. \tag{3.54}$$

S-flow The authors of [182] demonstrated that (3.54) can be parameterized by studying the time evolution of similarities S_i. More specifically, if averaging in (3.53) is relative to the e-connection on $\mathcal{W}$, then by [182, Proposition 3.6], the W-flow system (3.54) is equivalent to

$$\dot{W}(t) = \mathcal{R}_{W(t)}[S(t)], \qquad W(0) = \mathbb{1}_{\mathcal{W}} \tag{3.55a}$$

$$\dot{S}(t) = \mathcal{R}_{S(t)}[\Omega S(t)], \qquad S(0) = S(\mathbb{1}_{\mathcal{W}}). \tag{3.55b}$$

In particular, (3.55a) is determined completely by (3.55b), which itself can be defined independent of (3.55a). This motivates to consider *S-flow* dynamics

$$\dot{S}(t) = \mathcal{R}_{S(t)}[\Omega S(t)], \qquad S(0) = S_0 \tag{3.56}$$

as assignment flows in their own right. The authors of [182] further show that for symmetric Ω, (3.56) is the Riemannian ascent flow of the potential

$$J(S) = \frac{1}{2}\langle S, \Omega S\rangle \tag{3.57}$$

while W-flows (3.54) admit no such interpretation. Note that (3.57) mirrors the shape of linear-payoff replicator potentials (3.15) presented in Section 3.2. We return to details of this relationship in Chapter 4.

EGN Dynamics A property of both (3.56) and (3.54) is that averaging in local neighborhoods only explicitly constitutes interaction between node states. Interaction between different dimensions of a single node state is implicitly achieved through the geometric ramifications of simplex constraints on the state. A simple way of explicitly modeling interaction between dimensions of a single node state is by defining a matrix $B \in \mathbb{R}^{c\times c}$ and augmenting the S-flow vector field (3.56) by multiplication of $B^\top$ from the right

$$\dot{W}(t) = \mathcal{R}_{W(t)}[\Omega W(t) B^\top], \qquad W(0) = W_0. \tag{3.58}$$

Here and in the following, we again use the symbol W to denote state of assignment flows, as opposed to the symbol S used in (3.56). This serves to unify notation and to signal that (3.56) is an assignment flow in its own right, independent of its connection to (3.54) through the parameterization (3.55). (3.58) is the simplest instance of *evolutionary games on networks* (EGN) [133] which is more general than (3.56). To study dynamics like (3.58), it will be helpful to work with the operator

$$\mathrm{vec}_r \colon \mathbb{R}^{n\times c} \to \mathbb{R}^{nc} \tag{3.59}$$

which vectorizes matrices by stacking their rows. Further, define the operators $\mathcal{R}_W^{\mathfrak{v}}$, $\Pi_0^{\mathfrak{v}}$ and $\exp_W^{\mathfrak{v}}$ which apply to row-vectorized states $w = \mathrm{vec}_r(W)$, $v = \mathrm{vec}_r(V)$

$$\mathcal{R}_w^{\mathfrak{v}}[v] = \mathrm{vec}_r\Big(\mathcal{R}_W[V]\Big) \tag{3.60a}$$

$$\Pi_0^{\mathfrak{v}} v = \mathrm{vec}_r\Big(\Pi_0 V\Big) \tag{3.60b}$$

$$\exp_w^{\mathfrak{v}}(v) = \mathrm{vec}_r\Big(\exp_W(V)\Big). \tag{3.60c}$$

Then (3.58) can be written as

$$\dot{w}(t) = \mathcal{R}_{w(t)}^{\mathfrak{v}}[(\Omega \otimes B)w(t)], \qquad w(0) = \mathrm{vec}_r(W_0). \tag{3.61}$$

More generally, interaction does not need to be be uniform accross all graph edges. In this case, we define separate payoff matrices

$$B_{ik} \in \mathbb{R}^{c\times c}, \qquad ik \in \mathcal{E} \subseteq [n] \times [n] \tag{3.62}$$

for each edge and combine them to a matrix $\overline{A} \in \mathbb{R}^{nc\times nc}$ built from $n \times n$ blocks of size $c \times c$. If $ij \notin \mathcal{E}$, then the block at position ij is filled with zeros, otherwise, it is $B_{ik}^{\top}$. The resulting EGN dynamics read

$$\dot{w}(t) = \mathcal{R}_{w(t)}^{\mathfrak{v}}[\overline{A}w(t)], \qquad w(0) = \mathrm{vec}_r(W_0). \tag{3.63}$$

We return to this class of dynamics in Chapter 4.

4 Embedding Assignment Flows

As was already apparent in Chapter 3, assignment flows are closely related to replicator dynamics. In this chapter, we work to clarify the relationship in more detail.

Evolutionary game theory [95, 177] is an established framework for modeling problems in diverse areas ranging from mathematical biology [191, 192, 84, 152, 121] to economics [71, 176]. It assumes a dynamic perspective on games played by a large and well-mixed population of agents. In this context, the earliest dynamical model of population state is the *replicator equation* [199, 186], which has since been generalized in several ways [17, 51] to accommodate more complex situations.

Since the motivating task of assignment flows is to model interaction of multiple coupled random variables, the underlying state space is a product manifold of multiple probability simplices. In game theory, this situation corresponds to *multiple populations* of agents. Graph edges encode interaction between populations in the sense that agents randomly engage in games with opponents from their own population as well as from adjacent populations. Accordingly, one may call assignment flows *multi-population replicator dynamics*.

We will contrast multi-population dynamics with *multi-game dynamics*. The latter model agents on a single population that simultaneously play multiple games, earning cumulative payoff. The state space is a single simplex with dimension growing exponentially in the number of games. These situations have previously been studied by [87]. In particular, the autors find that interaction between games occurs whenever the population state is outside of a specific submanifold. Here, we specify a generalization of this submanifold and study the structure of resulting interaction, which turns out to be highly relevant to modelling complex joint distributions.

Our central tool of analysis is an embedding of multiple probability simplices into a combinatorially large simplex of joint distributions. This is closely related to *Segre embeddings* of projective spaces [77] which play a prominent role in many areas of mathematics and physics, such as *independence models* in algebraic statistics [64] and *entanglement* in quantum mechanics [19].

Based on this ansatz, we develop a geometric perspective and formalism to study the relationship between replicator dynamics of multiple populations and multi-game replicator dynamics. In particular, we demonstrate that the multi-game dynamics of [87] share a generic payoff structure with multi-population games. The proposed embedding also

constitutes a formal reduction of multi-population dynamics to – much higher-dimensional – single-population dynamics, which is helpful for theoretical analysis. We demonstrate this by transferring two results on the asymptotic behavior of replicator dynamics from the single-population to the multi-population setting.

Note that, between structured prediction with assignment flows and the study of biological systems, there is a growing need for more powerful dynamical models in emerging applications. For instance, [212] argue for the use of generalized replicator dynamics to model interactions in nature – considering multi-player interaction in a multi-game setting.

4.1 Embedding Formalism

Return again to the data labeling perspective introduced in Section 3.3. Direct enumeration of all c^n possible assignments of c classes to n graph nodes allows to frame data labeling as a single decision between $N = c^n$ label configurations. A relaxation of this descision problem can again be represented on a probability simplex. We call

$$\mathcal{S}_N, \qquad N = c^n \tag{4.1}$$

the *meta-simplex* of joint distributions. Relaxation of structured prediction on $\mathcal{S}_N$ is more general than relaxation on $\mathcal{W}$. This is because *every* joint distribution of n coupled discrete random variables is represented as *a single point* on $\mathcal{S}_N$. The price of this generality lies in high dimension. If n is large, states in $\mathcal{S}_N$ are numerically intractable, i.e. can not be represented as explicit probability vectors within practical memory constraints. Here, we treat the meta-simplex (4.1) as a means to build geometric intuition on complex joint distributions, working up to a novel representation of these combinatorial objects in Chapter 5. To this end, two closely related questions need to be answered.

1. How does the geometry of $\mathcal{W}$ relate to the geometry of $\mathcal{S}_N$?
2. How do assignment flows on $\mathcal{W}$ relate to replicator dynamics on $\mathcal{S}_N$?

Answering the first question turns out to provide illuminating perspective on the second. To this end, we propose an embedding of the assignment manifold $\mathcal{W}$ into the meta-simplex $\mathcal{S}_N$. This is based on the idea that every point $W \in \mathcal{W}$ can be regarded as marginals of a *factorizing* joint distribution of n discrete random variables.

The simplest nontrivial example is the case of two binary random variables X_1, X_2 with joint distribution $p \in \mathcal{S}_N$ ($n = 2$, $c = 2$, $N = 2^2 = 4$). The marginal distributions of X_1 and X_2 are

$$p_1 = \sum_{j \in [c]} p(\cdot, j) \in \mathcal{S}_c \tag{4.2a}$$

$$p_2 = \sum_{j \in [c]} p(j, \cdot) \in \mathcal{S}_c \tag{4.2b}$$

which we can collect as rows of a matrix

$$W = \begin{bmatrix} p_1^\top \\ p_2^\top \end{bmatrix} \in \mathcal{W}. \tag{4.3}$$

If X_1 and X_2 are independent, their joint distribution p factorizes into the marginal distributions (4.2)

$$p(j,l) = p_1(j)p_2(l). \tag{4.4}$$

We now generalize to arbitrary collections of discrete random variables. Let $p \in \mathcal{S}_N$ be a joint probability distribution of n discrete random variables $(X_1, \dots, X_n)$. For simplicity, we assume each random variable takes values in the same class set $[c]$, although the following results remain valid in more general scenarios. If p is viewed as a probability vector, each entry is the probability of a class configuration for the random variables. To manifest this fact in our formalism, we will use multi-indices $\gamma \in [c]^n$ to index p. More concretely, let $\gamma = (\gamma_1, \dots, \gamma_n) \in [c]^n$, then p_γ denotes the joint probability of the configuration $(X_1 = \gamma_1, \dots, X_n = \gamma_n)$.

Now consider the maps defined componentwise by

$$T\colon \mathcal{W} \to \mathcal{T} \subseteq \mathcal{S}_N, \qquad T(W)_\gamma := \prod_{i\in[n]} W_{i,\gamma_i} \qquad \text{for all } \gamma \in [c]^n \tag{4.5a}$$

$$Q\colon \mathbb{R}^{n\times c} \to \mathbb{R}^N, \qquad Q(X)_\gamma := \sum_{i\in[n]} X_{i,\gamma_i} \qquad \text{for all } \gamma \in [c]^n \tag{4.5b}$$

$$M\colon \mathbb{R}^N \to \mathbb{R}^{n\times c}, \qquad M(x)_{ij} := \sum_{\gamma\in[c]^n\,:\,\gamma_i=j} x_\gamma \qquad \text{for all } (i,j) \in [n]\times[c]. \tag{4.5c}$$

The particular choice of these maps will be justified by laying out several compatibility properties which intricately link them to each other and to the geometries of $\mathcal{W}$ and $\mathcal{S}_N$. Specifically,

- T represents factorizing joint distributions by their marginals, generalizing (4.4).
- T realizes the concept of enumerating configurations in the sense that the extremal points of $\overline{\mathcal{W}}$ are bijectively mapped to the extremal points of $\overline{\mathcal{S}_N}$.
- The restriction of M to $\mathcal{T}$ inverts T by computing node-wise marginals, generalizing (4.2). We choose the larger domain $\mathbb{R}^N$ for M such that it becomes the adjoint mapping of Q (Lemma 4.4).

The above choice of T is made to interpret the relationship between $\mathcal{W}$ and $\mathcal{S}_N$, answering the first question posed above. This choice is natural from a geometric standpoint, as the following theorem shows.

Theorem 4.1 (Assignment Manifold Embedding) *The map $T\colon \mathcal{W} \to \mathcal{T} \subseteq \mathcal{S}_N$ is an isometric embedding of $\mathcal{W}$ equipped with the product Fisher-Rao geometry, into $\mathcal{S}_N$ equipped with the Fisher-Rao geometry. On its image $T(\mathcal{W}) =: \mathcal{T} \subseteq \mathcal{S}_N$, the inverse is given by marginalization*

$$M|_{\mathcal{T}} = T^{-1}\colon \mathcal{T} \to \mathcal{W}. \tag{4.6}$$

Proof. A standard argument (Lemma A.1) shows that $T\colon \mathcal{W} \to \mathcal{T}$ is injective. We check

that the inverse of T has the shape (4.6).

$$(MT(W))_{i,j} = \sum_{\gamma\,:\,\gamma_i=j} \prod_{r\in[n]} W_{r,\gamma_r} = \sum_{\gamma\,:\,\gamma_i=j} W_{i,j} \prod_{r\in[n]\setminus\{i\}} W_{r,\gamma_r} \tag{4.7}$$

$$= \sum_{l\in[n]\setminus\{i\}} \sum_{\gamma_l\in[c]} \prod_{r\in[n]\setminus\{i\}} W_{r,\gamma_r} \tag{4.8}$$

$$= W_{i,j} \sum_{k_1\in[c]} W_{1,k_1} \sum_{k_2\in[c]} W_{2,k_2} \dots \sum_{k_n\in[c]} W_{n,k_n} \tag{4.9}$$

$$= W_{i,j} \prod_{r\in[n]\setminus\{i\}} \underbrace{\sum_{\gamma_r\in[c]} W_{r,\gamma_r}}_{=1} = W_{i,j}. \tag{4.10}$$

Clearly, all component functions of T and T^{-1} are smooth. We will now show that T is a topological embedding, i.e. a homeomorphism with respect to the subspace topology of $\mathcal{T} \subseteq \mathcal{S}_N$. Let

$$\mathcal{Q} = Q(T_0\mathcal{W}) \tag{4.11}$$

denote the image of $T_0\mathcal{W}$ under Q. $\mathcal{Q}$ is a linear subspace of $T_0\mathcal{S}_N$ because, for any $V \in T_0\mathcal{W}$, we have

$$QV = Q\Pi_0 V = \Pi_0 QV \in T_0\mathcal{S}_N \tag{4.12}$$

by Lemma A.3. In addition, Lemma A.5 shows $\ker Q \cap T_0\mathcal{W} = \{0\}$, since any matrix in $\ker Q$ has constant row vectors. Thus, the restriction of Q to $T_0\mathcal{W}$ is injective and since $T_0\mathcal{W}$ and $\mathcal{Q}$ have finite dimension, $Q|_{T_0\mathcal{W}}$ is a homeomorphism. The lifting map at the barycenter is the inverse of the *global* e-coordinate chart of information geometry up to a change of basis. In particular, $\exp_{\mathbb{1}_\mathcal{W}} : T_0\mathcal{W} \to \mathcal{W}$ and $\exp_{\mathbb{1}_{\mathcal{S}_N}} : T_0\mathcal{S}_N \to \mathcal{S}_N$ are homeomorphisms. Now let

$$\psi \colon \mathcal{T} \to \mathcal{Q}, \qquad p \mapsto \psi(p) = \exp^{-1}_{\mathbb{1}_{\mathcal{S}_N}}(p) \tag{4.13}$$

which is well-defined due to Lemma 4.3 and denote the initial topology of $\mathcal{T}$ with respect to ψ^{-1} by $\mathcal{A}$. Then T is a homeomorphism of $\mathcal{W}$ and $\mathcal{T}$ equipped with the topology $\mathcal{A}$ because

$$T = \exp_{\mathbb{1}_{\mathcal{S}_N}} \circ Q|_{T_0\mathcal{W}} \circ \psi^{-1} \tag{4.14}$$

by Lemma 4.3. It remains to show that $\mathcal{A}$ coincides with the subspace topology of $\mathcal{T} \subseteq \mathcal{S}_N$. Note that the topology of $\mathcal{Q}$ is the subspace topology of $\mathcal{Q} \subseteq T_0\mathcal{S}_N$ and recall that $\exp_{\mathbb{1}_{\mathcal{S}_N}} : T_0\mathcal{S}_N \to \mathcal{S}_N$ is a homeomorphism. For a subset $A \subseteq \mathcal{Q}$ we thus have

$$A \in \mathcal{A} \;\Leftrightarrow\; \psi(A) \text{ is open in } \mathcal{Q} \tag{4.15a}$$

$$\Leftrightarrow\; \exp^{-1}_{\mathbb{1}_{\mathcal{S}_N}}(A) = B \cap \mathcal{Q} \text{ for an open set } B \subseteq T_0\mathcal{S}_N \tag{4.15b}$$

$$\Leftrightarrow\; \exp^{-1}_{\mathbb{1}_{\mathcal{S}_N}}(A) = \exp^{-1}_{\mathbb{1}_{\mathcal{S}_N}}(\overline{A}) \cap \mathcal{Q} \text{ for an open set } \overline{A} \subseteq \mathcal{S}_N \tag{4.15c}$$

$$\Leftrightarrow\; A = \overline{A} \cap \exp_{\mathbb{1}_{\mathcal{S}_N}}(\mathcal{Q}) \text{ for an open set } \overline{A} \subseteq \mathcal{S}_N \tag{4.15d}$$

$$\Leftrightarrow\; A = \overline{A} \cap \mathcal{T} \text{ for an open set } \overline{A} \subseteq \mathcal{S}_N. \tag{4.15e}$$

This shows that $\mathcal{A}$ is the subspace topology of $\mathcal{T} \subseteq \mathcal{S}_N$ and thus, T is a topological embedding of $\mathcal{W}$ into $\mathcal{S}_N$.

We compute the rank of T by applying Lemma A.4. Let $W \in \mathcal{W}$ and $V \in T_0\mathcal{W}$ be in the kernel of $dT|_W$. Then

$$0 = dT|_W[V] = T(W) \diamond Q\Big[\frac{V}{W}\Big] \tag{4.16}$$

which implies $\frac{V}{W} \in \ker Q$ because $T(W)_\gamma \neq 0$ for all $\gamma \in [c]^n$. By Lemma A.5 this implies

$$V = W \diamond (\mathrm{Diag}(d)\mathbb{1}_{n\times c}) = \mathrm{Diag}(d)W \tag{4.17}$$

for some $d \in \mathbb{R}^n$ with $\langle d, \mathbb{1}_n\rangle = 0$. From $V \in T_0\mathcal{W}$ we find

$$0 = \langle V_i, \mathbb{1}_c\rangle = d_i\langle W_i, \mathbb{1}_c\rangle = d_i, \qquad \forall i \in [n] \tag{4.18}$$

which shows $V = 0$ by (4.17), i.e. $dT|_W$ has full rank. Thus, T is an injective immersion.

It remains to show that T is metric compatible. Suppose $W \in \mathcal{W}$ and $U, V \in T_0\mathcal{W}$ are arbitrary. Denoting the Fisher-Rao metric on $\mathcal{S}_N$ by $g^{\mathcal{S}_N}$ we get

$$\begin{aligned}
\big(T^* g^{\mathcal{S}_N}\big)_W(U,V) &= g^{\mathcal{S}_N}_{T(W)}(dT|_W[U], dT|_W[V]) && \text{(4.19a)}\\
&= \Big\langle dT|_W[U], \tfrac{\mathbb{1}}{T(W)} \diamond dT|_W[V]\Big\rangle && \text{(4.19b)}\\
&\overset{\text{(A.9)}}{=} \Big\langle dT|_W[U], Q\Big[\frac{V}{W}\Big]\Big\rangle && \text{(4.19c)}\\
&= \Big\langle M dT|_W[U], \frac{V}{W}\Big\rangle. && \text{(4.19d)}
\end{aligned}$$

Note that M is linear, implying $dM|_p = M$ for every $p \in \mathcal{S}_N$. Since M restricted to $\mathcal{T} = T(\mathcal{W})$ is the inverse of T, one has $M \circ T = \mathrm{id}_{\mathcal{W}}$. These two facts imply

$$M\Big[dT|_W[U]\Big] = dM|_{T(W)}\Big[dT|_W[U]\Big] = d\Big(M \circ T\Big)|_W[U] = d(\mathrm{id}_{\mathcal{W}})|_W[U] = U. \tag{4.20}$$

Plugging this result back into (4.19d) gives

$$\big(T^* g^{\mathcal{S}_N}\big)_W(U,V) = \Big\langle U, \frac{V}{W}\Big\rangle = g^{\mathcal{W}}_W(U,V) \tag{4.21}$$

which shows the assertion. □

In view of the expression (4.5a), it is clear that $\mathcal{T}$, the image of $\mathcal{W}$ under T, is precisely the set of rank-1 tensors in $\mathcal{S}_N \subseteq \mathbb{R}^N \cong (\mathbb{R}^c)^n$. The introductory example further clarifies, that $\mathcal{T}$ is the set of factorizing joint distributions. Since factorization of the joint distribution corresponds to independence of random variables, these distributions are the least informative among all joint distributions with the prescribed marginals.

Proposition 4.2 (Maximum Entropy) *For every $W \in \mathcal{W}$, the distribution $T(W) \in \mathcal{S}_N$ has maximum entropy among all $p \in \mathcal{S}_N$ subject to the marginal constraint $Mp = W$, with M given by* (4.5c).

Proof. We use the concepts of m-flat and e-flat submanifolds of information geometry, which justify applying the Pythagorean relation of information geometry. For details, we refer to [7]. The feasible set of all distributions with the prescribed marginals reads

$$\{T(W) + u \colon Mu = 0\} \cap \mathcal{S}_N \tag{4.22}$$

which is an m-flat submanifold of $\mathcal{S}_N$. In addition, Lemma 4.3 shows that $\mathcal{T}$ is an e-flat submanifold of $\mathcal{S}_N$. Let $p = T(W) + u$ denote an arbitrary feasible point. By (4.22) and Lemma 4.4 we have

$$\langle u, QV \rangle = \langle Mu, V \rangle = 0 \tag{4.23}$$

for all $V \in \mathbb{R}^{n \times c}$. Consider the m-geodesic connecting p with $T(W)$. It intersects $\mathcal{T}$ at $T(W)$ and we find

$$\langle dT|_W[V], u \rangle = \langle T(W) \diamond Q\Big[\frac{V}{W}\Big], u \rangle_{T(W)} = \langle Q\Big[\frac{V}{W}\Big], u \rangle = \langle \frac{V}{W}, Mu \rangle = 0 \tag{4.24}$$

by using Lemma A.4. With (4.24), m-flatness of (4.22) and e-flatness of $\mathcal{T}$ the prerequisites for the Pythagorean relation of information geometry [7, Theorem 3.8] are met. Using the cross-entropy $H(p, q) = -\langle p, \log q \rangle$ as well as the relative entropy $\mathrm{KL}(p, q) = \langle p, \log \frac{p}{q} \rangle$ and barycenter $\mathbb{1}_{\mathcal{S}_N} = \frac{1}{N}\mathbb{1}$, we find

$$\begin{aligned} H(T(W)) &= H(T(W), \mathbb{1}_{\mathcal{S}_N}) - \mathrm{KL}(T(W), \mathbb{1}_{\mathcal{S}_N}) \\ &= \log N - \mathrm{KL}(T(W), \mathbb{1}_{\mathcal{S}_N}) \end{aligned} \tag{4.25}$$

and consequently

$$\begin{aligned} H(p) &= H(p, \mathbb{1}_{\mathcal{S}_N}) - \mathrm{KL}(p, \mathbb{1}_{\mathcal{S}_N}) && (4.26)\\ &= \log N - \mathrm{KL}(p, \mathbb{1}_{\mathcal{S}_N}) && (4.27)\\ &= H(T(W)) + \mathrm{KL}(T(W), \mathbb{1}_{\mathcal{S}_N}) - \mathrm{KL}(p, \mathbb{1}_{\mathcal{S}_N}) && (4.28)\\ &\overset{(*)}{=} H(T(W)) + \mathrm{KL}(T(W), \mathbb{1}_{\mathcal{S}_N}) - \mathrm{KL}(p, T(W)) - \mathrm{KL}(T(W), \mathbb{1}_{\mathcal{S}_N}) && (4.29)\\ &= H(T(W)) - \mathrm{KL}(p, T(W)) && (4.30) \end{aligned}$$

by the Pythagorean relation $(*)$. Therefore $H(p) \leq H(T(W))$ with equality only for $p = T(W)$ which shows the assertion. □

In general, infinitely many joint distributions have the same collection of marginal distributions $W \in \mathcal{W}$. Proposition 4.2 shows that T precisely selects the least informative one among them. This situation is illustrated in Figure 4.1.

Theorem 4.1 expresses an intricate relationship between the product Fisher-Rao geometry of $\mathcal{W}$ and the regular Fisher-Rao geometry of $\mathcal{S}_N$. A similar compatibility is found between the lifting map (2.45) on $\mathcal{W}$ and its analog on $\mathcal{S}_N$.

Lemma 4.3 (Lifting Map Lemma) *Let $S \in \mathcal{W}$ and $V \in \mathbb{R}^{n \times c}$. Then the mappings T, Q given by* (4.5) *satisfy*

$$T\Big(\exp_S(V)\Big) = \exp_{T(S)}\Big(Q(V)\Big), \tag{4.31}$$

where $\exp_S$ on the left is given by (3.29) *and $\exp_{T(S)}$ on the right naturally extends the mapping* (2.45).

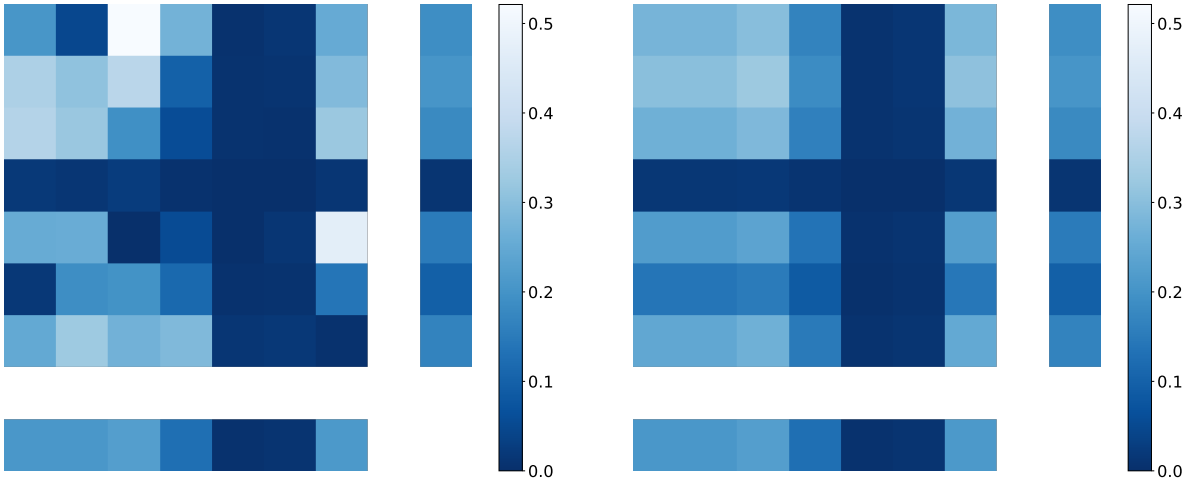

Figure 4.1: Marginals distributions $W = (W_1, W_2)$ and two possible conforming joint distributions. Joint distribution values are scaled by a factor of c for visual clarity. *Left:* A randomly generated joint distribution of W_1 and W_2. *Right:* The maximum-entropy joint distribution $T(W)$ of W_1 and W_2.

Proof. We have $T(\exp(V)) = \exp(Q(V))$ (without subscripts, i.e. applying the exponential function componentwise), because for any multi-index γ

$$\exp(Q(V))_\gamma = \exp\Big(Q(V)_\gamma\Big) = \exp\Big(\sum_{i\in[n]} V_{i,\gamma_i}\Big) \tag{4.32a}$$

$$= \prod_{i\in[n]} \exp\Big(V_{i,\gamma_i}\Big) = \prod_{i\in[n]} \Big(\exp(V)\Big)_{i,\gamma_i} \tag{4.32b}$$

$$= T(\exp(V))_\gamma. \tag{4.32c}$$

Let $D \in \mathbb{R}^{n\times n}$ be a diagonal matrix with nonzero diagonal entries. Then $T(DR) \propto T(R)$ for any $R \in \mathbb{R}^{n\times c}$ because

$$T(DR)_\gamma = \prod_{i\in[n]} (DR)_{i,\gamma_i} = \Big(\prod_{i\in[n]} D_{ii}\Big)\Big(\prod_{i\in[n]} R_{i,\gamma_i}\Big) \propto T(R)_\gamma. \tag{4.33}$$

It follows that

$$T(\exp_W(V)) \propto T(W \diamond \exp(V)) \overset{(4.32)}{=} T(W) \diamond \exp(Q(V)) \propto \exp_{T(W)}(Q(V)). \tag{4.34}$$

Because both the first and last term in (4.34) are clearly elements of $\mathcal{S}_N$, i.e. strictly positive vectors summing up ot 1, this implies the assertion. □

We will also frequently use the following useful identity connecting Q to marginalization.

Lemma 4.4 (Q Adjoint Lemma) *M and Q given by given by (4.5) are adjoint linear maps with respect to the standard inner product, i.e. for each $p \in \mathbb{R}^N$ and each $V \in \mathbb{R}^{n\times c}$ it holds that*

$$\langle p, Q(V)\rangle = \langle Mp, V\rangle. \tag{4.35}$$

Proof. Let $p \in \mathbb{R}^N$ and $V \in \mathbb{R}^{n \times c}$, then

$$\langle p, Q(V) \rangle = \sum_{\gamma \in [c]^n} p_\gamma Q(V)_\gamma = \sum_{\gamma \in [c]^n} p_\gamma \sum_{l \in [n]} V_{l,\gamma_l} = \sum_{l \in [n]} \sum_{j \in [c]} \sum_{\gamma_l = j} p_\gamma V_{l,\gamma_l} \tag{4.36a}$$

$$= \sum_{l \in [n]} \sum_{j \in [c]} V_{l,j} \sum_{\gamma_l = j} p_\gamma \overset{(4.5c)}{=} \sum_{l \in [n]} \sum_{j \in [c]} V_{l,j} (Mp)_{l,j} \tag{4.36b}$$

$$= \langle Mp, V \rangle. \tag{4.36c}$$

□

The central result of this chapter is stated next. It answers the second introductory question by demonstrating that the embedding $T \colon \mathcal{W} \to \mathcal{S}_N$ maps assignment flows on $\mathcal{W}$ to *single*-population replicator dynamics on $\mathcal{S}_N$ by a simple transformation of payoff functions.

Theorem 4.5 (Multi-Population Embedding Theorem) *For any payoff function $F \colon \mathcal{W} \to \mathbb{R}^{n \times c}$, the multi-population replicator dynamics*

$$\dot{W} = \mathcal{R}_W[F(W)], \quad W(0) = W_0 \tag{4.37}$$

on $\mathcal{W}$ is pushed forward by T to the replicator dynamics

$$\dot{p}(t) = R_{p(t)} \widehat{F}\Big(p(t)\Big), \quad p(0) = T(W_0), \qquad \widehat{F} = Q \circ F \circ M, \tag{4.38}$$

on $\mathcal{S}_N$.

Proof. We first show that, for any $W \in \mathcal{W}$, the differential of T and the replicator operator are related by

$$dT|_W[\mathcal{R}_W[X]] = R_{T(W)} Q[X], \quad \text{for all } X \in \mathbb{R}^{n \times c} \text{ and } W \in \mathcal{W}. \tag{4.39}$$

Let $\gamma \in [c]^n$ be an arbitrary multi-index. Because of $\mathcal{R}_W[X] \in T_0 \mathcal{W}$, Lemma A.4 implies

$$dT_\gamma|_W[\mathcal{R}_W[X]] = T_\gamma(W) Q_\gamma \left[\frac{\mathcal{R}_W[X]}{W} \right] = T_\gamma(W) \sum_{i \in [n]} \frac{(\mathcal{R}_W[X])_{i,\gamma_i}}{W_{i,\gamma_i}}. \tag{4.40}$$

Due to $(\mathcal{R}_W[X])_{i,\gamma_i} = W_{i,\gamma_i}(X_{i,\gamma_i} - \langle X_i, W_i \rangle)$, the sum can be written as

$$\sum_{i \in [n]} \frac{(\mathcal{R}_W[X])_{i,\gamma_i}}{W_{i,\gamma_i}} = \sum_{i \in [n]} \Big(X_{i,\gamma_i} - \langle X_i, W_i \rangle \Big) = Q_\gamma[X] - \langle X, W \rangle. \tag{4.41}$$

Additionally using the relation $W = M[T(W)]$ due to (4.6), and applying Lemma 4.4 gives

$$\langle X, W \rangle = \langle X, M[T(W)] \rangle = \langle Q[X], T(W) \rangle. \tag{4.42}$$

Collecting all expressions, we have

$$dT_\gamma|_W[\mathcal{R}_W[X]] = T_\gamma(W) \Big(Q_\gamma[X] - \langle Q[X], T(W) \rangle \Big) = \Big(R_{T(W)} Q[X] \Big)_\gamma \tag{4.43}$$

which shows (4.39). Now, denoting $p = T(W) \in \mathcal{S}_N$ we directly establish (4.38)

$$\dot{p} = dT(W)[\mathcal{R}_W[(F \circ M)(p)]] = R_p[(Q \circ F \circ M)(p)] = R_p[\widehat{F}(p)]. \tag{4.44}$$

□

Intuitively, the structure of $\widehat{F}$ in (4.38) can be seen as follows. The joint population state $p \in \mathcal{S}_N$ is first marginalized and payoff $F(Mp)$ is computed from the marginal multi-population state. Theorem 4.5 now shows that when multi-population state $W \in \mathcal{W}$ is seen as factorizing joint population state $p \in \mathcal{S}_N$ according to $p = T(W)$, then the payoff gained in state W is transformed by Q to induce replicator dynamics of the joint population state.

In the following, leading examples will be matrix games, i.e. linear payoff functions that model two-player interactions. Note, however, that Theorem 4.5 applies to arbitrary nonlinear payoff functions.

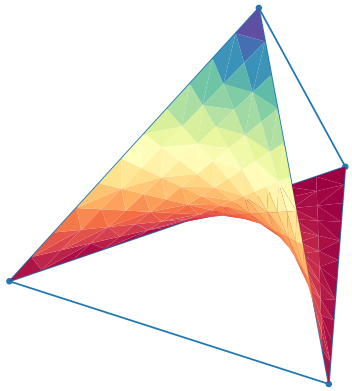

Figure 4.2: The embedded submanifold $\mathcal{T} \subseteq \mathcal{S}_N$. For two marginal distributions, this is known as the *Wright manifold* [95, Section 18.8], [40].

4.2 Multiple Populations and Multiple Games

Because both M and Q are linear operators, generalized matrix games on multiple populations reduce to simple matrix games of the joint population state exactly if the payoff F is a linear function of the multi-population state. Section 3.3.4 lists S-flows and EGN as examples of assignment flows which satisfy this criterion. Let

$$\overline{A} \in \mathbb{R}^{nc \times nc} \tag{4.45}$$

be an arbitrary payoff matrix for the vectorized state. Then by Lemma 4.4 and Theorem 4.5, the embedded dynamics in $\mathcal{T} \subseteq \mathcal{S}_N$ read

$$\dot{p}(t) = R_{p(t)}[Q\overline{A}Q^\top p(t)], \qquad p(0) = T(W_0). \tag{4.46}$$

Table 4.1 summarizes the scenarios discussed in the following, each special cases of (4.46) with different payoff matrices $\overline{A}$. We only list simple instances of each game for ease of exposition, making the structure of interaction more immediately apparent.

Table 4.1: Structure of payoff (4.45) for simple instances of different games.

	S-Flow	EGN	Multi-Game
Payoff	$\overline{A} = \Omega \otimes \mathbb{I}_c$	$\overline{A} = \Omega \otimes B^\top$	$\overline{A} = \mathbb{I}_n \otimes B^\top$

In particular, the *multi-game dynamics* of [87] can also be written as a matrix game in $\mathcal{S}_N$. Given matrices $A^{(i)} \in \mathbb{R}^{c\times c}$, $i \in [n]$, it is defined by

$$\dot{p}(t) = R_{p(t)}[Ap(t)], \qquad p(0) = p_0, \qquad A_{\alpha,\beta} = \sum_{i\in[n]} A^{(i)}_{\alpha_i,\beta_i}. \tag{4.47}$$

The following lemma shows that (4.47) has a simple shape within our formalism.

Lemma 4.6 *The payoff matrix in* (4.47) *can be written as* $A = Q\overline{A}Q^\top$ *where* $\overline{A}$ *denotes the block diagonal matrix with diagonal blocks* $A^{(i)}$*.*

Proof.

$$(Q\overline{A}Q^\top)_{\alpha,\beta} = \langle e_\alpha, Q\overline{A}Q^\top e_\beta\rangle = \langle Q^\top e_\alpha, \overline{A}Q^\top e_\beta\rangle \tag{4.48a}$$

$$= \sum_{i\in[n]} \langle e_{\alpha_i}, A^{(i)} e_{\beta_i}\rangle = \sum_{i\in[n]} A^{(i)}_{\alpha_i,\beta_i} \tag{4.48b}$$

□

In the simplest case, if all single-game payoff submatrices are the same $A^{(i)} = B \in \mathbb{R}^{c\times c}$, then multi-game dynamics have payoff $\overline{A} = \mathbb{I}_n \otimes B^\top$ as listed in Table 4.1. It was shown by [87] that the multi-game dynamics (4.47) do not generally decompose into individual single-game dynamics, unless the initialization is on the *Wright manifold* (see Figure 4.2). The set $\mathcal{T} \subseteq \mathcal{S}_N$ defined by (4.5a) is a generalization of the Wright manifold for $n > 2$ and Theorem 4.5 generalizes the decomposition of multi-game dynamics to more than two populations. For $p(0) \in \mathcal{T}$, the dynamics (4.47) are the embedded dynamics of

$$\dot{s}(t) = \mathcal{R}_{s(t)}[\overline{A}s(t)], \qquad s(0) = Mp(0) \tag{4.49}$$

by Lemma 4.6 and Theorem 4.5. For block diagonal $\overline{A}$, (4.49) is a collection of non-interacting single-game replicator dynamics

$$\dot{W}_i(t) = R_{W_i(t)}[A^{(i)}W_i(t)], \qquad W_i(0) = (Mp(0))_i, \qquad i \in [n] \tag{4.50}$$

in accordance with the findings of [87] for the specific case $n = 2$.

4.3 Tangent Space Embedding

Assignment flow evolve in the *curved* space $\mathcal{W}$ and the usual parameterization in ambient coordinates or m-coordinates of information geometry is subject to simplex constraints. For this reason, it is desirable in numerical computations to instead parameterize them in the tangent space $T_0\mathcal{W}$, which is a *flat* and *unconstrained* vector space. Recall from Theorem 3.3 that one such parameterization reads

$$W(t) = \exp_{\mathbb{1}_\mathcal{W}}(V(t)) \tag{4.51a}$$
$$\dot{V}(t) = \Pi_0 F(\exp_{\mathbb{1}_\mathcal{W}}(V(t))), \qquad V(0) = \exp_{\mathbb{1}_\mathcal{W}}^{-1}(W_0). \tag{4.51b}$$

With regard to the embedding Theorem 4.5, it turns out that analogous to the association of assignment matrices $W \in \mathcal{W}$ with factorizing joint distributions $T(W) \in \mathcal{S}_N$, Q assumes a corresponding role for tangent vectors in $T_0\mathcal{W}$.

Theorem 4.7 (Tangent Space Embedding Theorem) *The assignment flow tangent space dynamics*

$$\dot{V} = \Pi_0 F\Big(\exp_{\mathbb{1}_\mathcal{W}}(V)\Big), \quad V(0) = V_0 \tag{4.52}$$

on $T_0\mathcal{W}$ is pushed forward by Q to the tangent space replicator dynamics

$$\dot{U} = \Pi_0 \widehat{F}\Big(\exp_{\mathbb{1}_N}(U)\Big), \quad U(0) = Q(V_0), \qquad \widehat{F} = Q \circ F \circ M \tag{4.53}$$

on $T_0\mathcal{S}_N$.

Proof. Denoting $U = QV$ and using the lifting map (Lemma 4.3), we directly compute

$$\dot{U} = Q\dot{V} = Q\Pi_0 F\Big(\exp_{\mathbb{1}_\mathcal{W}}(V)\Big) \tag{4.54a}$$
$$= \Pi_0 QF\Big(\exp_{\mathbb{1}_\mathcal{W}}(V)\Big) \qquad \text{by Lemma A.3} \tag{4.54b}$$
$$= \Pi_0 QF\Big((M \circ T)\Big(\exp_{\mathbb{1}_\mathcal{W}}(V)\Big)\Big) \qquad \text{by (4.6)} \tag{4.54c}$$
$$= \Pi_0 QF\Big(M \exp_{\mathbb{1}_N}(QV)\Big) \qquad \text{by Lemma 4.3} \tag{4.54d}$$
$$= \Pi_0 QF\Big(M \exp_{\mathbb{1}_N}(U)\Big) \tag{4.54e}$$
$$= \Pi_0 \widehat{F}\Big(\exp_{\mathbb{1}_N}(U)\Big). \tag{4.54f}$$

□

Pushforward via Q thus preserves the shape of (4.52) up to the same fitness function transformation $\widehat{F} = Q \circ F \circ M$ from Theorem 4.5.

The set $\operatorname{img} Q \subseteq T_0\mathcal{S}_N$ contains exactly those tangent vectors corresponding to assignments $\mathcal{T} \subseteq \mathcal{S}_N$ via lifting, because $T(W) = T(\exp_{\mathbb{1}_\mathcal{W}})(V) = \exp_{\mathbb{1}_N}(QV)$ for any $W \in \mathcal{W}$ and $V = \exp_{\mathbb{1}_\mathcal{W}}^{-1}(W)$ by Lemma 4.3. In particular, U evolves in the linear subspace

$$\operatorname{img}\Big(\Pi_0 \circ Q\Big) \subseteq T_0\mathcal{S}_N. \tag{4.55}$$

This linear structure is reason to assume the study of tangent space flows (4.53) is easier than directly studying assignment flows (3.33) in some situations. In the latter case, the respective domain $\mathcal{T} \subseteq \mathcal{S}_N$ is the (curved) set of rank-1 tensors in $\mathcal{S}_N$.

Several applications have been proposed for the replicator dynamics of Section 4.2 including as a model of human brain functioning [134], collective learning [181], epileptic seizure onset detection [83], task mapping [135] and collective adaptation [180]. Assignment flows have been applied recently to the segmentation of digitized volume data under layer ordering constraints [190] as well as for unsupervised image labeling tasks, employing spatial regularization [231, 228]. These small sample of examples illustrate that replicator dynamics can act as powerful data models in diverse applications. In situations where only partial knowledge about the system is available, system parameters may also be learned from data as discussed in Section 3.3.3.

4.4 Asymptotic Behavior

A central topic in population dynamics is the study of how the properties of the underlying game characterized by the payoff function relate to steady states of the dynamical model. In this section, we describe how

- *Nash equilibria (NE)* and
- *Evolutionarily stable states (ESS)*

of multi-population games and their replicator dynamics behave under the embedding (4.5a). Nash equilibria for multi-population games are population states at which no agent (in any population) has payoff to gain from unilaterally switching strategies.

Definition 4.8 (Nash Equilibrium) *Let $\overline{\mathcal{W}}$, the closure of $\mathcal{W}$ be the set of multi-population states (n populations, c strategies) and let $F\colon \overline{\mathcal{W}} \to \mathbb{R}^{n\times c}$ be the payoff for a multi-population game. The set of Nash equilibria of F is defined as*

$$\mathrm{NE}(F) = \{W \in \overline{\mathcal{W}} \mid \forall i \in [n],\, \forall j \in \mathrm{supp}(W_i),\, \forall k \in [c] : F(W)_{i,j} \geq F(W)_{i,k}\} \tag{4.56}$$

Definition 4.8 naturally extends the classic notion of Nash equilibrium to multi-population games. Nash equilibria are preserved if the multi-population game is embedded as specified by Theorem 4.5.

Theorem 4.9 (Embedded Nash Equilibria) *Let $F\colon \overline{\mathcal{W}} \to T_0\overline{\mathcal{W}}$ be a multi-population game on $\overline{\mathcal{W}}$ and $\widehat{F} = Q \circ F \circ M$ be the related population game on $\mathcal{S}_N$. Then*

$$T\Big(\mathrm{NE}(F)\Big) = \mathrm{NE}(\widehat{F}) \cap \overline{\mathcal{T}}. \tag{4.57}$$

Proof. Let $W \in \mathrm{NE}(F)$ and let $\alpha \in \mathrm{supp}(T(W))$ be arbitrary. Then

$$\widehat{F}(T(W))_\alpha = (QF(W))_\alpha = \sum_{l\in[n]} F_{l,\alpha_l}(W) \tag{4.58a}$$

$$\geq \sum_{l\in[n]} F_{l,\beta_l}(W) = \widehat{F}(T(W))_\beta, \quad \forall\, \beta \in [c]^n, \tag{4.58b}$$

because $\alpha_l \in \text{supp}(W_l)$, $\forall l \in [n]$, by Lemma A.2 and W is a Nash equilibrium of F. This implies $T(\text{NE}(F)) \subseteq \text{NE}(\widehat{F}) \cap \overline{\mathcal{T}}$. Conversely, let $p \in \text{NE}(\widehat{F}) \cap \mathcal{T}$ have shape $p = T(W)$ and let $\alpha_l \in \text{supp}(W_l)$, $\forall l \in [n]$. Then $\alpha \in \text{supp}(p)$ by Lemma A.2 and

$$\sum_{l\in[n]} F_{l,\alpha_l}(W) = \widehat{F}(p)_\alpha \geq \widehat{F}(p)_\beta = \sum_{l\in[n]} F_{l,\beta_l}(W), \quad \forall\, \beta \in [c]^n, \tag{4.59}$$

because p is a Nash equilibrium of $\widehat{F}$. Choose $\beta \in [c]^n$ such that it matches α at all positions but $i \in [n]$. Then (4.59) implies $F_{i,\alpha_i}(W) \geq F_{i,\beta_i}(W)$ for arbitrary $\beta_i \in [c]$ which shows $\text{NE}(\widehat{F}) \cap \overline{\mathcal{T}} \subseteq T(\text{NE}(F))$. □

Definition 4.10 (Evolutionarily Stable State (ESS)) *A multi-population state $W^* \in \mathcal{W}$ is called an evolutionarily stable state (ESS) of a game $F\colon \mathcal{W} \to \mathbb{R}^{n\times c}$, if there is an environment $U \subseteq \mathcal{W}$ of W^* such that*

$$\langle W - W^*, F(W)\rangle < 0, \qquad \forall\, W \in U \setminus \{W^*\}. \tag{4.60}$$

This generalization of the classic ESS [191] to multi-population settings is called *Taylor ESS* by [177]. A property to recommend Definition 4.10 over the weaker notion of *monomorphic ESS* [50] is presented next.

Theorem 4.11 (Embedded ESS) *Let $F\colon \mathcal{W} \to \mathbb{R}^{n\times c}$ be a multi-population game. Then W^* is an ESS of F exactly if there exists an environment $U \subseteq \mathcal{T}$ of $T(W^*)$ such that*

$$\langle p - T(W^*), \widehat{F}(p)\rangle < 0, \qquad \forall\, p \in U \setminus \{T(W^*)\}, \tag{4.61}$$

where $\widehat{F} = Q \circ F \circ M$ denotes the embedded single-population game on $\mathcal{S}_N$ as specified by Theorem 4.5.

Proof. Since $U \subseteq \mathcal{T}$ and T is continuous, we may write $p = T(W)$ for W in an environment $M(U) \subseteq \mathcal{W}$ of W^*. (4.61) then reads

$$\begin{aligned}
\langle T(W) - T(W^*), \widehat{F}(T(W))\rangle &= \langle T(W) - T(W^*), QF(MT(W))\rangle && \text{(4.62a)}\\
&= \langle M(T(W) - T(W^*)), F(W)\rangle \text{ (Lemma 4.4)} && \text{(4.62b)}\\
&= \langle W - W^*, F(W)\rangle && \text{(4.62c)}
\end{aligned}$$

and the last row is strictly smaller than 0 for all $W \in M(U) \setminus \{W^*\}$ exactly if W^* is an ESS of F according to Definition 4.10. □

One useful aspect of Theorem 4.5 is that it formally reduces multi-population replicator dynamics to single-population ones. This enables us to transfer analysis of e.g. asymptotic behavior from the single-population to the multi-population setting. We first summarize standard results on the asymptotic behavior of replicator dynamics derived from a potential function and refer to [177] for a comprehensive overview.

Theorem 4.12 (Replicators converge to NE) *Let $\widehat{J}\colon \mathcal{S}_c \to \mathbb{R}$ be a C^1 potential such that the induced payoff function $\widehat{F} = \Pi_0 \nabla \widehat{J}$ is Lipschitz on $\mathcal{S}_c$. Then for any internal point $p_0 \in \mathcal{S}_c$, the replicator dynamics*

$$\dot{p}(t) = R_{p(t)}[\widehat{F}(p)], \qquad p(0) = p_0 \tag{4.63}$$

converge to a Nash equilibrium.

Proof. Because $\widehat{F}$ is Lipschitz, forward trajectories of the dynamics (4.63) are unique by the Picard-Lindelöf theorem. The potential $\widehat{J}$ is a strict Lyapunov function for replicator dynamics and unique forward trajectories converge to restricted equilibria [94, 178]. In general, there may exist restricted equilibria which are not Nash equilibria. In game theory, this property of replicator dynamics is called a lack of *Nash stationarity*. However, no internal trajectory converges to any of these points [28]. The solution trajectories of (4.63) are internal trajectories because p_0 is an internal point and $\mathcal{S}_c$ is invariant under all replicator dynamics with Lipschitz payoff function for finite time. The latter is clear from the tangent space parameterization of Theorem 3.3. □

There is a simple relationship between potential functions in the multi-population and single-population settings.

Lemma 4.13 (Potential Embedding) *If $F\colon \mathcal{W} \to T_0\mathcal{W}$ has potential J, then $\widehat{F} = Q \circ F \circ M$ has potential $\widehat{J} = J \circ M$.*

Proof. For $\widehat{J}(p) = (J \circ M)(p)$, we directly compute

$$\nabla \widehat{J}(p) = (DM(p))^\top \nabla J(W) = (M)^\top \circ \nabla J(W) = (Q \circ \nabla J \circ M)(p) \tag{4.64}$$

by denoting $W = M(p)$ and using Lemma 4.4. □

We can now use the embedded potential of Lemma 4.13 and embedded Nash equilibria of Theorem 4.9 to generalize the findings of Theorem 4.12 to multiple populations.

Theorem 4.14 (Multi-Population Replicators converge to NE) *Let $J\colon \mathcal{W} \to \mathbb{R}$ be a C^1 potential such that the induced payoff function $F = \Pi_0 \nabla J$ is Lipschitz on $\mathcal{W}$. Then, for any internal point $W_0 \in \mathcal{W}$, the multi-population replicator dynamics*

$$\dot{W}(t) = \mathcal{R}_{W(t)}[F(W)], \qquad W(0) = W_0 \tag{4.65}$$

converge to a Nash equilibrium.

Proof. Let $p(t) = T(W(t))$. Then $p(t)$ follows the single-population replicator dynamics (4.38) by Theorem 4.5 which are induced by the embedded potential $\widehat{J}(p) = J \circ M$ due to Lemma 4.13 and start at the interior point $T(W_0)$ of $\mathcal{S}_N$. By Theorem 4.12, $p(t)$ converges to a NE of $\widehat{F} = \Pi_0 \circ Q \circ F \circ M$ on $\mathcal{S}_N$. Since $p(t) = T(W(t)) \in \mathcal{T}$ for all times t, the limit point necessarily lies in the closure of $\mathcal{T}$. Theorem 4.9 then shows the assertion. □

By [178, Proposition 3.1] all Nash equilibria satisfy the KKT optimality conditions for maximizing J subject to simplex constraints. If J is concave, the KKT conditions are sufficient optimality conditions and thus (4.63) converges to a local maximizer. In addition, W is a Nash equilibrium exactly if $F(W)$ lies in the normal cone of the state space at W [86, 150]. Thus, convergence of (4.63) to a boundary point which is not an extremal point only occurs if the trajectory reaches the boundary exactly perpendicularly. For assignment flows, it has been known that convergence to a non-extremal point of $\mathcal{W}$ is an unusual occurrence. In fact, this behavior is not observed at all in the numerical solution of labeling problems for real-world data. [11] thus conjectured that convergence to

a non-extremal point only occurs for a null set of initial population states. This was shown to be true for non-negative, linear fitness functions derived from a quadratic potential [227], which we recalled as Theorem 3.2. From a game-theoretical perspective, only extremal points can be ESS under the posed conditions.

Note that the content of Theorem 4.14 is likely known to experts. We present this construction to illustrate the power of the formalism around Theorem 4.5, providing a mathematical toolset to reduce the analysis of multi-population replicator dynamics to single-population ones.

5 Discrete Joint Distributions

Recall the *data labeling* scenario described in Section 3.3. Given n data vectors, each from a metric space $\mathcal{F}$, the task is to infer a distribution of classes on n nodes. More formally, the central object of interest is the joint distribution

$$\mu^{\mathrm{LD}} \in \mathcal{P}(\mathcal{F}^n \times [c]^n) \tag{5.1}$$

of labeled data. In Section 3.3, we have focused on the conditional distributions

$$\mu(\cdot|x) \in \mathcal{P}([c]^n) \equiv \mathcal{S}_N \tag{5.2}$$

of labelings given data under the assumption that a labeling is deterministically associated with each collection of data vectors $x \in \mathcal{F}^n$, restricting $\mu(\cdot|x)$ to a discrete Dirac measure. Here, we go one step further and discuss representation and learning of *general* distributions $p \in \mathcal{S}_N$.

In applications, p may occur as a conditional measure (5.2). This is illustrated in Figure 5.1. The image on the left depicts an ambiguous subject. Each pixel of the image is associated with a node of $\mathcal{G}$, carrying the color of the pixel as a feature in some color space $\mathcal{F}$. The data distribution (5.1) is a distribution of semantically segmented images, i.e. each class in $[c]$ is a semantic class of image content (*cat*, *lion*, *background*, ...). If (5.2) were discrete Dirac distributions, each image would have a deterministic semantic segmentation. In the situation of Figure 5.1 however, the image can not be deterministically segmented, because there is ambiguity about the displayed subject. Suppose the subject could be *cat* or *lion* with equal probability. Then $\mu(\cdot|x)$ is a mixture of two equally weighted discrete Dirac distributions. We will discuss a specific approach to approximating distributions similar to this mixture in Section 5.2. Samples from our model are shown as segmentations in the first row. Even in this comparatively simple case, the distribution p of interest does not factorize into marginals. Recalling the notions of Chapter 4, there does not exist a $W \in \mathcal{W}$ such that $p = T(W)$. An approximation is the product of marginal distributions $T(Mp)$. Samples from this distribution are shown in the second row of Figure 5.1. Clearly, this does not capture the nature of the sought data distribution, because it is unable to properly represent the dependencies between graph nodes.

[1]This image was created by DALL-E 2 [166].

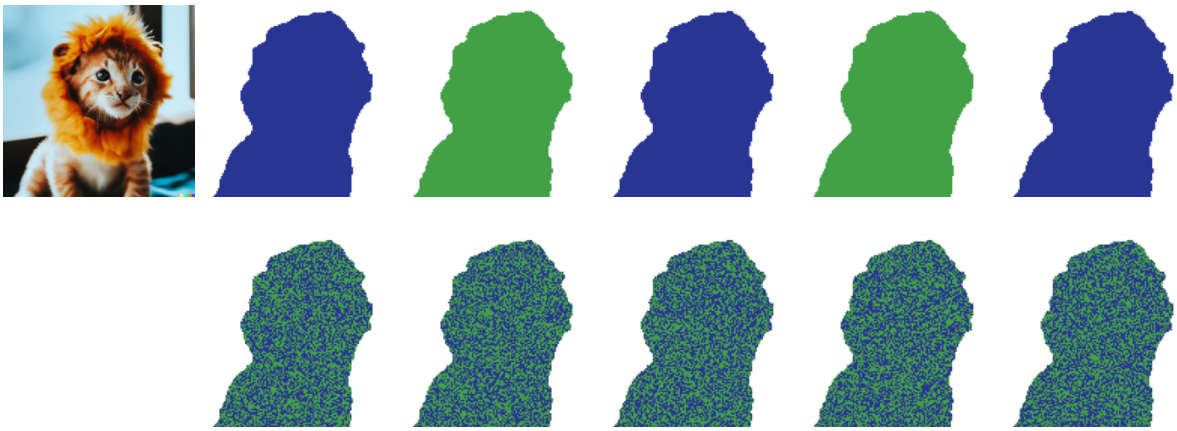

Figure 5.1: Image segmentation of an ambiguous subject[1](cat or lion with equal probability). *First row:* Samples from our approximation of the data distribution $p \in \mathcal{S}_N$, which couples subject pixels. *Second row:* Samples from, the product of marginal distributions $T(Mp)$, which amounts to a pixel-wise independence assumption and therefore fails to represent spatial context.

The distribution $p \in \mathcal{S}_N$ of interest may also occur directly as a data distribution, without associated features. An example are generative models of natural-language text. Suppose text is tokenized with a vocabulary of size c and consider token sequences of length n. In practice, n may be called *context length* in the sense that dependencies between tokens are only modelled if they occur at distance at most n from each other in a corpus. The joint distribution of tokens in a context window of size n is a distribution $p \in \mathcal{S}_N$. This p is a data distribution of interest with no associated features.

Probabilistic models for context-sensitive decision making and structured prediction have been a focal point of research during the last decades. Major paradigms for representing complex probability distributions include Gibbs distributions, probabilistic graphical models [214, 144] and measure transport using push-forward maps parameterized by neural networks [173, 109]. In this chapter, we discuss a modelling approach for representation and learning of complex joint distributions based on assignment flows. As outlined in Chapter 3, assignment flow approaches combine probabilistic state spaces akin to relaxation domains of probabilistic graphical models with parameterized dynamical systems akin to the ones used for measure transport.

Any probability distribution over discrete variables can be represented as a point in the meta-simplex $\mathcal{S}_N$. The embedded submanifold $\mathcal{T} = T(\mathcal{W})$ contains all factorizing distributions. We leverage the geometric understanding of factorizing distributions built in Chapter 4 to enable efficient parameterization of larger classes of joint distributions which *do not factorize*. The basic underlying idea is that $\mathcal{T}$ is *curved* and thus, means of points in $\mathcal{T}$ with respect to m-geometry of $\mathcal{S}_N$ are mixture distributions which generally lie outside of $\mathcal{T}$. We summarize the properties of $\mathcal{T}$ underlying this construction in the following lemma.

Lemma 5.1 (Shape of $\mathcal{T}$) *$\mathcal{T}$ is an e-flat submanifold of $\mathcal{S}_N$ which is non-convex as a subset of $\mathbb{R}^N$.*

Proof. The lifting map lemma 4.3 gives

$$T(\exp_{\mathbb{1}_\mathcal{W}}(V)) = \exp_{\mathbb{1}_{\mathcal{S}_N}}(QV), \qquad \forall V \in T_0\mathcal{W}. \tag{5.3}$$

Since $\exp_{\mathbb{1}_\mathcal{W}} \colon T_0\mathcal{W} \to \mathcal{W}$ is surjective onto $\mathcal{W}$, (5.3) characterizes $\mathcal{T} = T(\mathcal{W})$ as the image of the linear subspace $\operatorname{img} Q \subseteq T_0\mathcal{S}_N$ under $\exp_{\mathbb{1}_{\mathcal{S}_N}} \colon T_0\mathcal{S}_N \to \mathcal{S}_N$. Thus, $\mathcal{T}$ is flat in e-coordinates on $\mathcal{S}_N$. The subspace $\operatorname{img} Q$ has dimension at most $n(c-1)$ because Q is linear and $T_0\mathcal{W}$ has dimension $n(c-1)$. To see that $\mathcal{T}$ is not convex, note that the extreme points of $\overline{\mathcal{W}}$ are bijectively mapped to the extremal points of $\overline{\mathcal{S}}_N$ by T. Suppose $\mathcal{T}$ was convex. Then $\mathcal{T}$ contains the convex hull of every subset of $\mathcal{T}$. But the convex hull of the extremal points of $\mathcal{S}_N$ contains all of $\mathcal{S}_N$, contradicting the fact that $\mathcal{T}$ has lower dimension than $\mathcal{S}_N$. □

Since $\mathcal{S}_N$ is a convex set, every point of $\mathcal{S}_N$ can be represented as a mixture of extremal points. These extremal points in turn correspond to discrete Dirac distributions and all of them lie in the closure of $\mathcal{T}$. Thus, in principle, every distribution in $\mathcal{S}_N$ can be represented as a mixture of points in (the closure of) $\mathcal{T}$. However, such a representation may require a combinatorial number of mixture components, making it intractable.

To alleviate this issue, we propose to model points $p \in \mathcal{S}_N$ as means over parameterized continuous distributions $T_\sharp\nu \in \mathcal{P}(\mathcal{T})$. There are many possible ways of parameterizing the underlying $\nu \in \mathcal{P}(\mathcal{W})$. Here, we consider two approaches based on measure transport via assignment flows. The first is a randomized ODE: a distribution over parameters is learned, each generating a different ODE. The second is a normalizing flow: a deterministic dynamical system is learned which transports a reference distribution of initial states over time. In Section 5.1, we introduce both methods. In Section 5.2, we approximate a given energy-based model by learning a randomized assignment flow. Here, convergence to discrete Dirac measures is crucial to the underlying entropy approximation and we restrict the learned dynamics accordingly to ensure convergence. To achieve this restriction, we work with a randomized ODE approach. In Section 5.3, we turn to the problem of approximating data distributions which are only accessible through a dataset of samples. In this case, convergence to discrete Dirac distributions is a second priority and we choose the normalizing flow approach which restricts the learned vector fields as little as possible.

Over the last decades, a range of variational approximations [22] have been developed for the problems of inference and parameter learning when modelling joint distributions of coupled random variables. The most basic one, the so-called ('naive') mean-field approximation [214, Section 5], minimizes relative entropy between a fully factorized distribution and the intractable target distribution. More advanced structured mean-field approaches include the well-known Bethe- and Kikuchi approximations and related algorithms for approximating marginals of the target distribution by belief propagation [155], [214, Section 4], convexified Bethe approximations [213, 89] and related methods in statistical physics, like the cavity method [170]. Since our approach is based on a geometric understanding of factorizing distributions, the mean-field approximation is a natural first point of comparison.

We point out that our approach to probabilistic modeling and inference, by convex combination of extreme points of compact convex sets of probability distributions, is not at all new in mathematics, but in fact extends far beyond the scenarios with discrete random variables considered here [65]. However, our approach to constructing these representations is novel.

5.1 Parameterization of Joint Distributions

Our goal is to model the joint distribution $p \in \mathcal{S}_N$ of $n > 0$ discrete random variables, each taking values in $[c]$. The embedding $T\colon \mathcal{W} \to \mathcal{S}_N$ introduced in Chapter 4 enables to define $q \in \mathcal{S}_N$ conditioned on an assignment matrix $W \in \mathcal{W}$ by the identification

$$q(\gamma|W) = T(W)_\gamma, \qquad \gamma \in [c]^n. \tag{5.4}$$

We can then make the ansatz

$$q_{\mathfrak{p}}(\gamma, W) = q(\gamma|W)\nu_{\mathfrak{p}}(W) \tag{5.5}$$

for some distribution $\nu_{\mathfrak{p}} \in \mathcal{P}(\mathcal{W})$ parameterized by $\mathfrak{p}$. Marginalization yields the model distribution

$$q_{\mathfrak{p}}(\gamma) = \int q_{\mathfrak{p}}(\gamma, W)dW = \int q(\gamma|W)\nu_{\mathfrak{p}}(W)dW = \mathbb{E}_{W\sim\nu_{\mathfrak{p}}}[T(W)_\gamma]. \tag{5.6}$$

As was outlined above, the parameterization (5.6) can in principle represent any joint distribution of discrete random variables. In practice, its representation ability depends on the chosen parameterization of $\nu_{\mathfrak{p}} \in \mathcal{P}(\mathcal{W})$. Recall the most general definition of assignment flows with parameterized payoff function

$$\dot{W}(t) = \mathcal{R}_{W(t)}\Big[F_{\mathfrak{p}}\Big(W(t)\Big)\Big], \qquad W(0) = W_0 \tag{5.7}$$

from Section 3.3 as well as the flow map $\psi_{\mathfrak{p}}(W_0, t)$ generated by (5.7). The following sections describe two parameterizations of $\nu_{\mathfrak{p}}$ which build on (5.7).

5.1.1 Randomized Assignment Flows

Suppose we have chosen a parameterized shape of payoff function $F_{\mathfrak{p}}\colon \mathcal{W} \to \mathbb{R}^{n\times c}$. If the parameters $\mathfrak{p}$ are taken to be a random variable with distribution ρ on some parameter space, this turns $\psi_{\mathfrak{p}}(W_0, t)$ into a random variable. We now fix an initialization $W_0 \in \mathcal{W}$ and define $\nu_{\mathfrak{p}} \in \mathcal{P}(\mathcal{W})$ as the distribution of $\psi_{\mathfrak{p}}(W_0, t)$. Since this distribution still varies over time $t > 0$, we denote it $\nu_{\mathfrak{p}}(t)$.

If we fix a finite integration time $0 < t < \infty$, drawing samples from $q_{\mathfrak{p}}$ amounts to drawing a set of parameters $\mathfrak{p} \sim \rho$, integrating (5.7) numerically to compute $W_t = \psi_{\mathfrak{p}}(W_0, t)$ and finally drawing a sample $\gamma \in [c]^n$ from the categorial distribution $T(W_t)$. In contrast, if we use infinite integration time $t \to \infty$ and we can guarantee that (almost) every draw of parameters $\mathfrak{p} \sim \rho$ generates an integral curve $\{\psi_{\mathfrak{p}}(W_0, t)\}_{t>0}$ that approaches an extremal

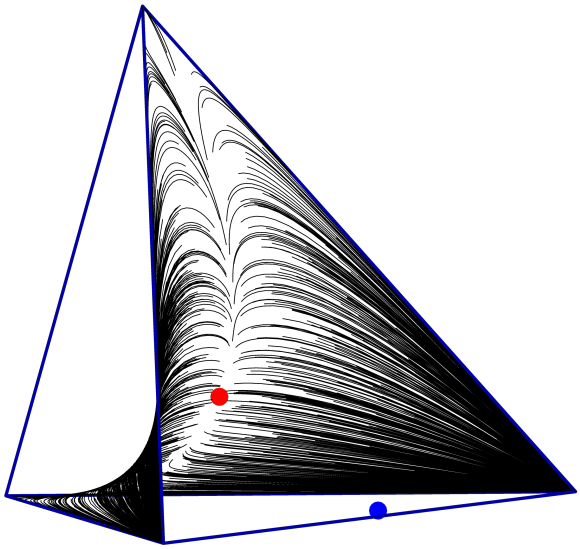

Figure 5.2: Visualization of 1000 samples from the target distribution (*blue point*). Each sample is associated with an embedded integral curve $T(W(t))$ (4.5a) of the assignment flow ODE (5.7) on the Wright manifold of factorizing distributions $\mathcal{T} \subseteq \mathcal{S}_4$. The flow pushes forward a standard Gaussian reference distribution on the tangent space $T_0\mathcal{W}$ at the barycenter (*red point*), which is lifted to $\mathcal{T}$ and transported to extremal points, corresponding to class configurations via (3.21).

point of $\mathcal{W}$, then the second sampling step becomes trivial due to the correspondence (3.21) of extremal points with discrete Dirac distributions. A specific choice of payoff function and parameter distribution which meets these requirements is specified next.

Theorem 5.2 (Convergence of Embedded S-Flow) *Let* $\Omega = \max(Z+Z^\top, 0)+\epsilon\mathbb{1}_n$, $Z \in \mathbb{R}^{n\times n}$, $\epsilon > 0$, *where the entries of* Z *follow a multivariate normal distribution and maximization is componentwise. Then the embedded S-flow*

$$\dot{p}(t) = R_{p(t)}[Q\Omega M p(t)], \qquad p(0) = T(W_0) \tag{5.8}$$

converges to a discrete Dirac measure $\delta_{\gamma(\Omega)}$ *for every draw of* Ω *and almost every initialization* $W_0 \in \mathcal{W}$.

Proof. For the given shape of Ω, the assumptions of Theorem 3.2 ([227, Theorem 2]) are met, which guarantees that the solution $W(t)$ of the S-flow (3.56) converges to an integral solution for almost every initialization $W_0 \in \mathcal{W}$. Because T bijectively maps the corners of $\mathcal{W}$ to the corners of $\mathcal{S}_N$, the assertion then follows from the embedding theorem 4.5. □

5.1.2 Continuous Normalizing Assignment Flows

An established method for representing complex multimodal distributions is by applying a parameterized bijective transformation to a tractable reference measure [173, 109]. The

flow of an ordinary differential equation is a bijective mapping on its domain by the Picard-Lindelöf theorem, provided that the driving vector field satisfies a Lipschitz condition. This is the underlying idea of continuous normalizing flows (CNFs) [43], which employ vector fields defined by deep neural networks. The state space is usually Euclidean and the reference distribution is typically chosen as a normal distribution, which justifies the name 'normalizing flow'. In order to parameterize a distribution $\nu_{\mathfrak{p}} \in \mathcal{P}(\mathcal{W})$, we can first define a simple reference measure $\nu_0^n \in \mathcal{P}(\mathcal{W})$ as the distribution of initializations W_0 for (5.7). To this end, first choose an orthonormal basis of the tangent space $T_0\mathcal{S}$ and define the linear map

$$H\colon \mathbb{R}^{n(c-1)} \to T_0\mathcal{W} \tag{5.9}$$

which associates coordinates in this basis with tangent vectors. Now define

$$\nu_0 = (\exp_{\mathbb{1}_{\mathcal{W}}} \circ H)_\sharp \mathcal{N}_0 \tag{5.10}$$

for standard normal distribution $\mathcal{N}_0 \in \mathcal{P}(\mathbb{R}^{n(c-1)})$. The reference measure ν_0^n is simple in the sense that sampling and likelihood evaluation under ν_0^n are numerically tractable.

After fixed time $t > 0$, a parameterized model distribution is found as

$$\nu_{\mathfrak{p}} = \psi_{\mathfrak{p}}(\cdot, t)_\sharp \nu_0^n \in \mathcal{P}(\mathcal{W}). \tag{5.11}$$

The parameters $\mathfrak{p}$ in (5.11) are *not* a random variable as in Section 5.1.1. They can be chosen freely as parameters of a payoff function $F_{\mathfrak{p}}$ driving the dynamics (5.7). This allows for much flexibility in the choice of payoff functions, including deep neural networks. We call the measure transport approach (5.11) *continuous normalizing assignment flows* (CNAF).

Figure 5.2 shows samples from a CNAF as integral curves on $\mathcal{T} \subseteq \mathcal{S}_N$. It represents a target distribution (blue) in $\mathcal{S}_N$ which is the joint distribution of two strongly coupled binary random variables. The parametrized payoff function $F_{\mathfrak{p}}$ of (5.7) is trained in a stable and efficient way by matching e-geodesic curves on the assignment manifold. Details of the training procedure will be discussed in Section 5.3.3.

5.2 Approximation of Energy-based Models

As a first application, we consider the approximation of energy-based models. Here, the probability of every configuration $\gamma \in [c]^n$ is given by

$$p_\gamma = \frac{1}{Z^*} \exp(-\lambda E_\gamma), \qquad Z^* = \sum_{\gamma \in [c]^n} \exp(-\lambda E_\gamma). \tag{5.12}$$

Inverse temperature λ and *energy* E_γ of each individual configuration is assumed to be tractable but the *partition function* Z^* is intractable, because it contains a combinatorially large number of summands. We enumerate the energies of all configurations and collect them in the single vector $E \in \mathbb{R}^N$. As an instance of (5.12), consider the class of pairwise graphical models with energy

$$E_\gamma = \sum_{i \in [n]} \langle \mathfrak{p}^i, e_{\gamma_i} \rangle + \sum_{ij \in \mathcal{E}} \langle e_{\gamma_j}, \mathfrak{p}^{ij} e_{\gamma_i} \rangle \qquad \mathfrak{p}^i \in \mathbb{R}^c, \quad \mathfrak{p}^{ij} \in \mathbb{R}^{c \times c} \tag{5.13}$$

Tying back to the notation of earlier sections, we transform the pairwise term in (5.13) to

$$\sum_{ij\in\mathcal{E}} \langle e_{\gamma_j}, \mathfrak{p}^{ij} e_{\gamma_i}\rangle = \sum_{ij\in\mathcal{E}} \langle (Q^\top e_\gamma)_j, \mathfrak{p}^{ij} (Q^\top e_\gamma)_i\rangle = \langle Q^\top e_\gamma, \mathfrak{p}^{(p)} Q^\top e_\gamma\rangle \tag{5.14}$$

with matrix $\mathfrak{p}^{(p)} \in \mathbb{R}^{nc\times nc}$ built from blocks $\mathfrak{p}^{ij} \in \mathbb{R}^{c\times c}$, $i, j \in [n]$. Similarly, combining unary parameters $\mathfrak{p}^i$ into a single vector $\mathfrak{p}^{(u)} \in \mathbb{R}^{nc}$ yields the vectorized form of (5.13)

$$E = Q\mathfrak{p}^{(u)} + \operatorname{diag}(Q\mathfrak{p}^{(p)} Q^\top). \tag{5.15}$$

Suppose one approximates p by a tractable $q \in \mathcal{S}_N$. This entails minimization of

$$\begin{aligned} \mathrm{KL}(q, p) &= \left\langle q, \log \frac{q}{p} \right\rangle = \langle q, \log q\rangle - \langle q, \log p\rangle && \text{(5.16a)} \\ &= -H(q) - \lambda\langle q, E\rangle + \underbrace{\log Z^*}_{\text{const}} \underbrace{\langle P, \mathbb{1}_N\rangle}_{=1} && \text{(5.16b)} \end{aligned}$$

which mirrors the well-known conjugacy relation [38, Lemma 1.1.3]

$$\log \left\langle \mathbb{1}_{\mathcal{S}_N}, \exp(-\lambda E)\right\rangle = \sup_q \; -\lambda\langle E, q\rangle - \mathrm{KL}(q, \mathbb{1}_{\mathcal{S}_N}). \tag{5.17}$$

Note that, while $\mathrm{KL}(p, q) = 0$ exactly if $p = q$, relative entropy is not a symmetric measure for the difference of both arguments. Thus, the order of arguments in (5.16) influences the behavior of the approximation. If the model distribution is in the second argument, the approximation tends to favor covering the mass of p. Each mode will be covered but possibly not matched very precisely. If the model distribution is in the first argument as in (5.16), the approximation tends to favor matching the most prominent modes of p, at the expense of not covering other modes. See [146] for a detailed treatment of this phenomenon, including generalization to α-divergences. The order of arguments in (5.16) is not chosen with the goal of inducing mode-seeking behavior. Instead, the choice is made purely for tractability of the involved quantities. If we switch the order of arguments

$$\mathrm{KL}(p, q) = \left\langle p, \log \frac{p}{q} \right\rangle = -H(p) - \langle p, \log q\rangle, \tag{5.18}$$

relative entropy involves the log-likelihood of q in expectation under p, which can not easily be computed for energy-based models.

Thus, in order to learn q efficiently, we opt to use (5.16) which involves the model entropy and expected energy as well as their respective gradients. Since the energy of each individual configuration is tractable, the expected energy of a tractable model is typically easy to estimate. However, estimating entropy from samples is generally a difficult problem, which makes tractable entropy a key design criterium for the choice of q. Along this line of reasoning, the basic *mean-field approach* is to approximate p by a factorizing distribution $T(W)$. The model entropy in (5.16) then simplifies to

$$-H(T(W)) = \langle T(W), \log T(W)\rangle = \langle T(W), Q \log W\rangle = \langle W, \log W\rangle \tag{5.19}$$

by Lemma 4.4. Because both the barycenter $\mathbb{1}_{\mathcal{S}_N}$ of $\mathcal{S}_N$ and every extremal point of $\mathcal{S}_N$ is a factorizing distribution, the mean-field approximation generally works best if either (a) all configurations have close to the same probability or (b) p is close to a discrete Dirac distribution. The first scenario is called *high temperature* regime in statistical physics. High temperature systems $1 \gg \lambda > 0$ are dominated by randomness. Even if all modes of p were known, they are not reliable indicators of system state due to dominating randomness. In contrast, the second scenario pertains to systems which are essentially deterministic. It is the challenging medium or low temperature regime in which a more sophisticated model is typically required – entailing the problem of entropy estimation.

5.2.1 Energy and Entropy

We now aim to approximate p by $q_{\mathfrak{p}}$ as defined in (5.6). To parameterize $\nu_{\mathfrak{p}} \in \mathcal{P}(\mathcal{W})$, we choose the randomized assignment flow approach of Section 5.1.1. The reason for this choice lies in the necessity to approximate model entropy. To achieve an effective estimator, we will leverage convergence to discrete Dirac distributions by Theorem 5.2 which is easier to guarantee for randomized assignment flows as opposed to CNAF (Section 5.1.2).

Expected model energy reads

$$\langle q_{\mathfrak{p}}, E \rangle = \langle \mathbb{E}_{W \sim \nu_{\mathfrak{p}}} T(W), E \rangle = \mathbb{E}_{W \sim \nu_{\mathfrak{p}}} \langle T(W), E \rangle \tag{5.20}$$

which amounts to an expected value of mean field energies. Thus, if mean field energy is tractable, the empirical energy over samples $W \sim \nu_{\mathfrak{p}}$ is an unbiased estimator of model energy.

We turn to the more challenging problem of entropy estimation. Typically, estimating model entropy $H(q)$ from samples is difficult because the support $|\operatorname{supp} q| = s$ of q is large compared to the number m of available samples. The support of p can be arbitrarily large in principle. In fact, as a prerequisite for the Hammersley-Clifford theorem [34, Thm. 9.1.10], full support has formal merit in Markov random fields. On the other hand, many situations of practical interest do not benefit from a model with very large support. For instance, in image segmentation, most configurations of classes on the pixels of an image will have very little semantic content. In statistical mechanics, full support is beneficial to model high temperature systems. However, as mentioned above, the behavior of these systems is dominated by randomness and they are well-described by a mean field approximation. In contrast, for the challenging medium or low temperature regime, small support can result in a good approximation.

Suppose the support size s is small compared to the number m of available samples $\{\gamma^{(k)}\}_{k \in [m]}$ drawn from $q_{\mathfrak{p}}$. Denote by

$$\widehat{q} = \frac{1}{m} \sum_{k \in [m]} \delta_{\gamma^{(k)}} \in \mathcal{S}_N \tag{5.21}$$

the empirical distribution of these samples. A classical analysis by [145] shows that the *plugin estimator*

$$H(q) \approx H(\widehat{q}) = - \sum_{\gamma \in \operatorname{supp}(\widehat{q})} \widehat{q}_\gamma \log \widehat{q}_\gamma \tag{5.22}$$

has bias

$$\mathbb{E}[H(\widehat{q})] - H(q) = -\frac{s-1}{2m} + \mathcal{O}\Big(\frac{1}{m^2}\Big) \tag{5.23}$$

which leads to the Miller-Maddows bias correction for known support s.

It was shown that the bias-corrected estimator still only achieves consistency if $m \gg s$ [156] which is far from the optimal rate of $m \gg s/\log s$ [101]. More advanced approaches also exist [208, 101, 215, 209] which are likely able to achieve better sample efficiency relative to the support size s. Note that the support of the model distribution q defined in (5.6) is typically not known, making it difficult to correct the bias of plugin estimators according to (5.23) or to judge the effectiveness of estimating entropy in this fashion.

In our experiments, we use the support of the empirical distribution (5.21) as a surrogate. For the problem instances we tested, relatively small support is observed and we still achieve good approximation of the target distribution. Thus, for the cases studied here, the estimator (5.22) with bias correction (5.23) appears sufficient. Note that an unbiased estimator of entropy from samples exists [148] but is not practical for our use case, because it entails drawing an indeterminate number of samples.

In order to learn parameters $\mathfrak{p}$, we would also like an approximation of $-H(\widehat{q})$ to be differentiable. Suppose, all samples W^k drawn from $q_{\mathfrak{p}}$ were extremal points of $\mathcal{W}$. Then $T(W^k) = \delta_{\gamma(k)} \in \mathcal{S}_N$ for some $\gamma(k) \in [c]^n$. The latter discrete Dirac distribution can equivalently be written as a unit vector $e_{\gamma(k)} \in \mathbb{R}^N$. We then find

$$-H(\widehat{q}) = \Big\langle \frac{1}{m}\sum_{k\in[m]} T(W^k), \log \frac{1}{m}\sum_{k\in[m]} T(W^k) \Big\rangle \tag{5.24a}$$

$$= \Big\langle \frac{1}{m}\sum_{k\in[m]} e_{\gamma(k)}, \log \frac{1}{m}\sum_{k\in[m]} e_{\gamma(k)} \Big\rangle = \frac{1}{m}\sum_{k\in[m]} \log\Big(\frac{1}{m}\sum_{l\in[m]} e_{\gamma(l)}\Big)_{\gamma(k)} \tag{5.24b}$$

$$= -\log m + \frac{1}{m}\sum_{k\in[m]} \log\Big(\sum_{l\in[m]} e_{\gamma(l)}\Big)_{\gamma(k)} \tag{5.24c}$$

which motivates the approximation

$$-H(\widehat{q}) \approx -\log m + \frac{1}{m}\sum_{k\in[m]} \log\Big(\sum_{l\in[m]} T(W^l)\Big)_{\gamma(k)} \tag{5.25}$$

for general samples $W^l \in \mathcal{W}$, i.e. not necessarily extremal points. The assumption of samples being extremal points results in sparsity of the sum (5.24b). This is crucial for the development of numerical methods, because general probability vectors in $\mathcal{S}_N$ can not be represented numerically without association of some underlying low-dimensional quantity. Thus, in order for the entropy approximation (5.25) to become exact, we need to use parameterized measures $\nu_{\mathfrak{p}}$ which concentrate on extremal points of $\mathcal{W}$. This is ensured by convergence of the transporting assignment flow to extremal points.

Note that $T(W^l)_{\gamma(k)}$ above is a product of n numbers in $(0,1)$. We thus rewrite the

summands in (5.25) as

$$\log\Big(\sum_{l\in[m]} T(W^l)\Big)_{\gamma(k)} \overset{(4.5a)}{=} \log\Big(\sum_{l\in[m]}\prod_{i\in[n]} W^l_{i,\gamma(k)_i}\Big)_{\gamma(k)} \tag{5.26a}$$

$$= \log\sum_{l\in[m]}\exp\Big(\sum_{i\in[n]}\log W^l_{i,\gamma(k)_i}\Big) \tag{5.26b}$$

to avoid numerical underflow problems by leveraging a stabilized implementation of the logsumexp function. Also note that the right-hand side is differentiable.

Once a suitable approximation of p is found by minimizing (5.16) with respect to $\mathfrak{p}$, the model $q_\mathfrak{p}$ can be used for probabilistic inference. Marginal distributions are easily estimated via

$$Mq_\mathfrak{p} = \mathbb{E}_{W\sim\nu_\mathfrak{p}}[MT(W)] = \mathbb{E}_{W\sim\nu_\mathfrak{p}}[W]. \tag{5.27}$$

Further, the expectation of any quantity $Q\phi$ with $\phi\in\mathbb{R}^{nc}$ under the learned model can be inferred by

$$\mathbb{E}_{q_\mathfrak{p}}[Q\phi] = \langle\mathbb{E}_{W\sim\nu_\mathfrak{p}}[T(W)], Q\phi\rangle = \mathbb{E}_{W\sim\nu_\mathfrak{p}}[\langle W,\phi\rangle]. \tag{5.28}$$

If no such structure is available, any probabilistic inference task $\mathbb{E}_{q_\mathfrak{p}}[f]$ can still be approached by drawing samples $\{W^k\}_{k\in[m]}$ from $\nu_\mathfrak{p}$, which correspond to system configurations $\{\gamma^k\}_{k\in[m]}\subseteq[c]^n$ by convergence to extremal points. An unbiased estimator is then given as

$$\mathbb{E}_{q_\mathfrak{p}}[f] \approx \frac{1}{m}\sum_{k\in[m]} f(\gamma^k). \tag{5.29}$$

5.2.2 Experiments

The introductory example in Fig. 5.1 was produced by approximating a Potts model [10, 164] on the grid graph of image pixels. This was achieved by randomizing EGN dynamics (3.63), giving $\overline{A}\in\mathbb{R}^{nc\times nc}$ the structure of multi-channel convolution with weights following a multivariate normal distribution. Suitable moments for this normal distribution together with a suitable flow initialization s_0 are the result of a *training procedure* which minimizes (5.16). To this end, a reparameterization trick [107] is applied in conjunction with the approximation (5.25) and bias correction (5.23) where the unknown support s is replaced by the empirical support $\widehat{s} = |\operatorname{supp}\widehat{q}|$ smoothed by the mean entropy of node-wise assignment. Numerical integration of (3.63) via the simple geometric Euler scheme [225] (step size 0.1, end time 1.0) is unwound and automatically differentiated by PyTorch [158] which allows to find a local optimum of parameters by employing the Adam optimizer [106] with step length 0.01. This experiment demonstrates two aspects of the proposed methodology. First, learning of randomized assignment flows easily scales to image data, even if we only use a simple *discretize-then-optimize* approach to compute stochastic gradients. Second, even though the distribution of parameters $\mathfrak{p}$ was not constrained to guarantee convergence of the dynamics (3.63) to extremal points, this behavior is still often observed in practice and the entropy approximation (5.25) did result in sufficient gradient precision for learning.

We further demonstrate the approximation of energy-based models on a small *two-dimensional Ising model*, i.e. a system of binary random variables with nearest-neighbor interaction on a grid graph, governed by a Gibbs distribution of the form (5.12) with a corresponding energy function E. These systems are classical ones in statistical mechanics [159]. They prototypically represent a combinatorially large configuration space and *long range* correlation at low temperatures. As a consequence, in the presence of an 'external field' [15], i.e. data defining unary potentials, minimizing energy and probabilistic inference become computationally difficult, even for moderate problem sizes. Such models initiated research on image segmentation and Bayesian inference [73, 75] and have been stimulating research on variational approximations for many years [214, 144]. As a consequence, they define an ideal testbed for empirical validation of our approach.

$\mathcal{G}$ is chosen as a 3×8 grid graph such that the combinatorial partition function and true marginals can still be computed by brute force. This allows to give numerical values for the distance to the combinatorial model in terms of relative entropy via (5.16). The number of classes is $c = 2$. Unary energy is chosen as -3.0 for the 0-configuration of nodes on the left boundary and as 3.0 on the right boundary. All other unary energies are zero. Pairwise energy is set to $\mathfrak{p}^{ij} = \frac{7}{10} \cdot (\mathbb{1}_c \mathbb{1}_c^\top - \mathbb{I}_c)$ for each edge.

We approximate this model by the same training procedure as above with reduced learning rate $5 \cdot 10^{-3}$ over 5k iterations. This takes around 21 minutes on a single desktop graphics card. To guarantee S-flow convergence via Theorem 5.2, we omit label interaction as afforded by EGN dynamics (3.63) and instead use S-flow dynamics (3.56) with symmetric matrix $\Omega \in \mathbb{R}^{n \times n}$ parameterized as $\Omega = \max(Z + Z^\top, 0) + 10^{-3}\mathbb{I}_n$ and entries of $Z \in \mathbb{R}^{n \times n}$ following a multivariate normal distribution. We initialize the distribution of Z centered at $\frac{1}{20}\mathbb{1}_n\mathbb{1}_n^\top$ and with componentwise variance 10^{-1}. In the early stages of optimization, samples are not integral due to the finite time horizon, but we observe that the sample entropy gradually decreases over the course of optimization, making the approximation (5.25) already close to exact for finite time. Once a model is learned through convergence to a local minimum, samples are guaranteed to approach extremal points of $\mathcal{W}$ for $t \to \infty$ by Theorem 5.2. In fact, it was shown in [227] that the same integer limit is also found by rounding after sufficiently large but finite time t which is relevant for numerical implementation.

As a baseline, we compute a mean field approximation $W \in \mathcal{W}$ by parameterizing $W = \mathrm{softmax}(V)$ and using the Adam optimizer to learn V by minimizing the tractable form of (5.16). This procedure is repeated for 1k initializations drawn randomly from a standard normal distribution of $V \in \mathbb{R}^{n \times c}$ and a model with minimal KL distance is selected from resulting local optima. The true distribution has multiple modes, of which mean field approximation can only represent a single one. In contrast, our model is able to capture the multimodality as is apparent from samples (Fig. 5.3), close approximation of marginals (Fig. 5.4) and low relative entropy (Tab. 5.1).

In the low temperature regime, the mass of p concentrates around its modes. For this reason, the proposed model – for which small support is computationally beneficial – actually becomes more effective at lower temperature. This unusual performance characteristic makes our approach promising in challenging structured prediction scenarios where mean-field approximation fails to capture the structure of interest.

Figure 5.3: Samples from approximated Ising model. *First row:* the mean-field baseline. *Second row:* our model. This demonstrates that, unlike the mean-field approximation, our approach can explore multiple modes in the low-temperature regime.

Figure 5.4: Marginals of the true distribution (*left*), our approximation via randomized assignment (*middle*) and the baseline mean-field approximation (*right*).

Table 5.1: Summary of Ising model approximation. Relative entropy to the true distribution is computed by brute-force evaluation of the combinatorial partition function. Entropy of our model is closely approximated by (5.22) with bias correction (5.23) using $m = 1\text{M}$ integer samples.

Model	KL	Energy	Entropy	Marginal Difference
AF (ours)	**0.599**	-1.98	2.56	**0.090**
Mean Field	1.974	-1.57	1.60	0.198

5.3 Approximating Empirical Data Distributions

We now turn to the problem of approximating a distribution $p \in \mathcal{S}_N$, which does not have a known form as an energy-based model, but a dataset of m samples $\{\gamma_k\}_{k\in[m]}$, independently drawn from p is given instead. In this case, the order of arguments in the relative entropy (5.16) chosen in the previous section

$$\mathrm{KL}(q, p) = -H(q) - \langle q, \log p\rangle \tag{5.30}$$

is no longer tractable, because it requires unknown log-likelihood under p. However, we can now estimate

$$\mathrm{KL}(p, q) = -H(p) - \langle p, \log q\rangle = -H(p) - \mathbb{E}_p[\log q] \tag{5.31}$$

by replacing the expected value on the right hand side with a mean over samples. This requires that the model distribution q has a density with respect to the Lebesgue measure and that log-likelihood of data under the model can be optimized for the purpose of learning q.

5.3.1 Continuous Normalizing Flows

An established approach to address the issue of constructing models with tractable likelihood is generative modeling with *normalizing flows* [167, 109, 157, 173]. Suppose we are working on a Euclidean domain $\mathbb{R}^d$ and want to represent a measure $q = (\psi_\mathfrak{p})_\sharp \mathcal{N}_0 \in \mathcal{P}(\mathbb{R}^d)$ by pushforward of a reference (normal) distribution $\mathcal{N}_0 \in \mathcal{P}(\mathbb{R}^d)$ under some invertible, parameterized transport function $\psi_\mathfrak{p}\colon \mathbb{R}^d \to \mathbb{R}^d$. By a change of variables [24], log-likelihood of $x \in \mathbb{R}^d$ under q can be written as

$$\log q(x) = \log p(\psi_\mathfrak{p}^{-1}(x)) - \log\det d\psi_\mathfrak{p}(x). \tag{5.32}$$

To build an effective model, we need $\psi_\mathfrak{p}$ to be efficiently differentiable and invertible. Further, since the goal is to learn complex multi-modal distributions, we seek a class of functions which has high capacity as a data model subject to these requirements. One popular direct construction are affine coupling architectures [60]. Here, we focus on the high-capacity class of invertible mappings on $\mathbb{R}^d$ defined through neural ordinary differential equations (nODEs) [43]. Any Lipschitz function $f_\mathfrak{p}\colon \mathbb{R}^d \to \mathbb{R}^d$ can be employed to define a vector field with invertible flow map. Evaluation of the inverse flow map is then performed efficiently through numerical integration backward in time. The *instantaneous change of variables* formula [43, Appendix A] clarifies likelihood computation (5.32) in this scenario, which we recite as the following theorem.

Theorem 5.3 (Instantaneous Change of Variables) *Let $x(0)$ be a random variable with distribution $\mathcal{N}_0 \in \mathcal{P}(\mathbb{R}^d)$ and let*

$$\dot{x}(t) = f(x(t)) \tag{5.33}$$

be an ODE defining time evolution on $\mathbb{R}^d$*. Let* $\psi(\cdot, t)\colon \mathbb{R}^d \to \mathbb{R}^d$ *denote the flow of* (5.33) *and* q_t *denote the density of* $\psi(\cdot, t)_\sharp \mathcal{N}_0$ *with respect to the Lebesgue measure on* $\mathbb{R}^d$*. If* f *is uniformly Lipschitz continuous, then*

$$\frac{\partial}{\partial t} \log q_t(x) = -\operatorname{tr}\Big(df(x)\Big) \tag{5.34}$$

for all $t \geq 0$ *and* $x \in \mathbb{R}^d$*.*

For fixed integration time $t > 0$, Theorem 5.3 can be used to compute the likelihood of $x \in \mathbb{R}^d$ under $q = q_t \in \mathcal{P}(\mathbb{R}^d)$ as

$$\log q_t(x) = \log q_0(x(0)) - \int_0^t \operatorname{tr}\Big(df(x(\tau))\Big) d\tau \tag{5.35}$$

where q_0 is the density of $\mathcal{N}_0$ with respect to the Lebesgue measure on $\mathbb{R}^d$ and $\{x(\tau)\}_{\tau\in[0,1]}$ is the unique integral curve of (5.33) with $x(1) = x$. A way to perform numerical evaluation of (5.35) is by quadrature of the integral. In order to compute the required integrand values, we first integrate the trajectory $x(t)$ backward in time, starting from $x(1) = x$. After reaching $x(0)$, the log-density of $\mathcal{N}_0$ at $x(0)$ is available in closed form. It remains to find a way of efficiently evaluating the Jacobian trace $\operatorname{tr}\Big(df(x)\Big)$ for general functions f. In particular, if f is a deep neural network and the underlying space is high dimensional $d \gg 1$, exact evaluation of the Jacobian trace is comutationally expensive. To remedy this, [78] propose to use Hutchinson's trace estimator [99].

Lemma 5.4 (Hutchinson's Trace Estimator) *Let* $A \in \mathbb{R}^{d\times d}$ *and* $\xi \in \mathcal{P}(\mathbb{R})$ *be any distribution with mean zero and unit variance. Then*

$$\mathbb{E}_{v\sim\xi^d}\langle v, Av\rangle = \operatorname{tr} A. \tag{5.36}$$

Typical examples for ξ in Lemma 5.4 are standard normal or Rademacher distribution. Applying Lemma 5.4 to estimate the Jacobian trace yields

$$\mathbb{E}_{v\sim\xi^d}\langle v, df(x)v\rangle = \operatorname{tr} df(x). \tag{5.37}$$

In practice, if f is a deep neural network, the Jacobian action $df(x)v$ can be computed efficiently by a backward pass through the network. Only a single sample $v \sim \xi^d$ is typically used to approximate the expected value in (5.37) during training. Note that automatic differentiation of this estimator with respect to parameters of f is still possible in modern deep learning software, even though evaluation of (5.37) itself requires a backward pass. For instance, PyTorch [158] can automatically create a compute graph for backpropagation through f and subsequently differentiate with respect to parameters by traversing the graph. This ability is crucial for learning continuous normalizing flows in the described manner, because both *discretize-then-optimize* and the adjoint sensitivity method described in Theorem 3.4 require the respective gradient.

Returning to (5.31), continuous normalizing flows make a parametric ansatz $f_\mathfrak{p}$ for the vector field in (5.33), usually a deep neural network, and learn $\mathfrak{p}$ by minimizing

$$\mathrm{KL}(p, q_\mathfrak{p}) = -H(p) - \mathbb{E}_p[\log q_\mathfrak{p}] \approx -H(p) - \frac{1}{m}\sum_{k\in[m]} [\log q_\mathfrak{p}(x_k)] \tag{5.38}$$

via stochastic gradient-based methods for a dataset $\{x_k\}_{k\in[m]}$ of independent samples drawn from the target distribution p. The target distribution entropy $H(p)$ is typically not known. However, since $H(p)$ does not depend on the parameters $\mathfrak{p}$, it can be treated as a constant when learning $q_{\mathfrak{p}}$.

5.3.2 Dequantization

While (5.31) applies to *discrete* data distributions $p \in \mathcal{S}_N$ in principle, it is not immediately clear how (5.38) is most effectively optimized in this scenario. If we make the ansatz (5.6) for $q_{\mathfrak{p}}$, log-likelihood reads

$$\log q_{\mathfrak{p}}(\gamma) = \log \mathbb{E}_{W\sim\nu_{\mathfrak{p}}}[T(W)_\gamma]. \tag{5.39}$$

Suppose the data distribution of interest contains significant coupling between the variables of different nodes. A strong model for $\nu_{\mathfrak{p}}$ should then concentrate probability mass at extremal points of $\mathcal{W}$, on distributions with small support. To understand this, return to the introductory example of Figure 5.1. Subject pixels are strongly coupled – if one of them is assigned the class *cat*, it is very likely that all other subject pixels are also assigned *cat*. A factorizing distribution $T(W)$ can only produce samples with this behavior, if W is essentially an extremal point of $\mathcal{W}$. This is because factorization into marginals amounts to independent distributions on each node. Unless $T(W)$ is already close to deterministic, it represents weak coupling and noisy samples like in the second row of Figure 5.1. However, complex target distributions p with large diversity of samples are supported on a large subset of $[c]^n$. Here, *large* does not necessarily refer to a large fraction of the full support size c^n, but to a large number relative to the size of a sample batch that can practically be drawn from $\nu_{\mathfrak{p}}$. The combination of these factors leads to difficult optimization of (5.39). For any batch $(W_1, \dots, W_{m_b}) \sim \nu_{\mathfrak{p}}^{m_b}$ of size m_b, the probability

$$\mathbb{P}_{\gamma\sim p}\Big(\gamma \in \bigcup_{k\in[m_b]} \operatorname{supp} T(W_k)\Big) \tag{5.40}$$

is small if W_k are extremal points of $\mathcal{W}$ and $m_b \ll |\operatorname{supp} p|$, leading to poor training signal in (5.38).

A way around this problem is to approximate discrete data distributions by continuous ones through *dequantization*. To this end, choose a latent space $\mathcal{F}^n$ and an embedding of class configurations $\gamma \in [c]^n$ as prototypical points $f_\gamma^* \in \mathcal{F}^n$. Suppose the choice of these points is fixed before training and associate disjoint sets $A_\gamma \subseteq \mathcal{F}^n$ with class configurations such that they form a partition of $\mathcal{F}^n$ and $f_\gamma^* \in A_\gamma$. We can then define the continuous surrogate model

$$\rho = \sum_{\gamma\in[c]^n} p_\gamma \mathcal{U}_{A_\gamma} \in \mathcal{P}(\mathcal{F}^n) \tag{5.41}$$

which represents $p \in \mathcal{S}_N$ as a mixture of uniform distributions, supported on the disjoint subsets A_γ. The underlying idea is that

$$\mathbb{P}_\rho(A_\gamma) = \int_{A_\gamma} \rho(y)dy = p_\gamma \int_{A_\gamma} \mathcal{U}_{A_\gamma}(y)dy = p_\gamma \tag{5.42}$$

due to the disjoint support of mixture components in (5.41). Denote a continuous model distribution by $\nu \in \mathcal{P}(\mathcal{F}^n)$. Using Jensen's inequality, we find

$$\begin{aligned} -H(\rho) - \mathrm{KL}(\rho, \nu) &= \int \rho(y) \log \nu(y) dy && \text{(5.43a)} \\ &= \sum_{\gamma \in [c]^n} p_\gamma \int_{A_\gamma} \log \nu(y) dy && \text{(5.43b)} \\ &\leq \sum_{\gamma \in [c]^n} p_\gamma \log \int_{A_\gamma} \nu(y) dy && \text{(5.43c)} \\ &= -H(p) - \mathrm{KL}(p, q) && \text{(5.43d)} \end{aligned}$$

for the discrete model distribution q defined by

$$q_\gamma = \int_{A_\gamma} \nu(y) dy = \mathbb{P}_\nu(A_\gamma). \tag{5.44}$$

Thus, fitting ν to ρ by maximizing log-likelihood of smoothed data drawn from ρ implicitly minimizes an upper bound on the relative entropy $\mathrm{KL}(p, q)$. In practice, drawing smoothed data from ρ amounts to adding noise to the prototypes $f^*_{\gamma_k} \in \mathcal{F}^n$ of discrete data $\{\gamma_k\}_{k \in [m]}$.

A slightly simpler version of the above construction was first proposed by [201]. The authors focus on image data which, although originally continuous, are only available discretized into 8-bit integer color values for efficient digital storage. In this case, the underlying continuous color imparts a natural structure on the set of discrete classes. Similar colors are naturally represented as prototypes which are close to each other with respect to some metric on the feature space $\mathcal{F}^n$. Similarly, it is apparent how to find low-dimensional Euclidean feature spaces with this structure. The necessity for dequantization in this setting has been noted by [207] who observe that continuous models reach artificially high likelihood by concentrating mass close to the discrete prototypical points. The construction of [201] justifies the previously known heuristic of adding noise to dequantize data. This has since become common practice for training normalizing flows on image data [60, 175, 8] and was generalized to non-uniform noise distributions by [92]. The latter work points out that (5.43) learns a continuous distribution with constant density on each region A_γ. The authors argue that this is an unnatural target for smooth model distributions and propose a second, conditional measure transport model for the noise distribution to be learned jointly.

For more general discrete data, it is still desirable to represent structure on the set of classes when embedding into a latent space, but this structure may not be apparent to human eyes. As a remedy, [41] present an approach to learning the embedding jointly with likelihood maximization. They subsequently define the partition of $\mathcal{F}^n$ into subsets A_γ through Voronoi tesselation. A special case of the dequantization method with direct connection to assignment flows is given by the choice of $\mathcal{F}^n = \mathcal{S}^n_c = \mathcal{W}$. The CNAF parameterization of $\nu \in \mathcal{P}(\mathcal{W})$ proposed in Section 5.1.2 can be combined with Voronoi tesselation of $\overline{\mathcal{W}}$. However, the resulting model distribution

$$q^{\mathrm{r}}_{\mathfrak{p}}(\gamma) = \mathbb{P}_{W \sim \nu_{\mathfrak{p}}}(W \in A_\gamma). \tag{5.45}$$

according to (5.44) differs from the ansatz

$$q_{\mathfrak{p}}(\gamma) = \mathbb{E}_{W\sim\nu_{\mathfrak{p}}}[T(W)_\gamma] \tag{5.46}$$

proposed in Section 5.1, unless all samples $W \sim \nu_{\mathfrak{p}}$ are extremal points of $\mathcal{W}$. Recall that sampling from the model (5.46) amounts to drawing $W \sim \nu_{\mathfrak{p}}$ and subsequently drawing a class configuration $\gamma \sim T(W)$. In contrast, drawing a sample from (5.45) can be performed by *rounding* W to the nearest extremal point, which itself is identified with a class configuration γ through (3.21). This is because the region $A_\gamma \subseteq \mathcal{W}$ associated with the labeling γ is the Voronoi cell with anchor point $M\delta_\gamma \in \overline{\mathcal{W}}$. For extremal points $W \sim \nu_{\mathfrak{p}}$, $T(W)$ is a discrete Dirac distribution and thus, all samples drawn from $T(W)$ match the class configuration γ associated via (3.21). This clarifies that (5.46) is equivalent to (5.45) if $\nu_{\mathfrak{p}}$ is a mixture of Dirac distributions on extremal points of $\mathcal{W}$.

The ability to learn an embedding of class configurations as prototypical points f^*_γ in a latent space, thereby representing similarity relations between classes, is central to the approach of [41]. This motivates the question whether such relationships between classes can also be learned if, as in our approach, the underlying state space $\mathcal{W}$ and (extremal) points associated with discrete class configurations are fixed. Because points in $\mathcal{S}_c$ have a clear interpretation as categorical distributions, we are able to achieve this goal by extending the payoff function $F_{\mathfrak{p}}$ as follows. For some $L > 0$, let $E \in \mathbb{R}^{L\times c}$ be a learnable embedding matrix. The columns of E can be seen as prototypical points in the Euclidean latent space $\mathbb{R}^L$. The action of E on extremal points $e_j \in \mathcal{S}_c$ precisely selects one of these columns, associating it with the class $j \in [c]$. Learning E now allows to represent relationships between classes in the latent space $\mathbb{R}^L$. Let $\mathcal{E}\colon \mathbb{R}^{n\times c} \to \mathbb{R}^{n\times c}$ denote the linear operator which applies E node-wise. We now choose a parameterized function $\widetilde{F}_{\mathfrak{p}}\colon \mathbb{R}^L \to \mathbb{R}^L$ that operates on $\mathbb{R}^L$ and define the payoff function

$$F_{\mathfrak{p}} = \mathcal{E}^\top \circ \widetilde{F}_{\mathfrak{p}} \circ \mathcal{E}\colon \mathcal{W} \to \mathbb{R}^{n\times c}. \tag{5.47}$$

Learning E jointly with other parameters $\mathfrak{p}$ amounts to learning class relationships in the latent space $\mathbb{R}^L$.

In order to work within the natural geometry of $\mathcal{W}$ established and motivated in Chapter 3, but simultaneously stay close to the Euclidean state spaces predominantly underlying CNF methods, we can consider the tangent space dynamics equivalent to assignment flows (5.7) according to Theorem 3.3. The regions $A_\gamma \subseteq \mathcal{W}$ correspond to convex cones in the tangent space

$$\widetilde{A}_\gamma = \{V \in T_0\mathcal{W}\colon V_{i,\gamma_i} \geq V_{i,j}\ \forall i \in [n],\ j \neq \gamma_i\} \tag{5.48}$$

up to overlap of these sets which has vanishing Lebesgue measure in $T_0\mathcal{W}$. A complication associated with viewing CNAF through this lense is that extremal points of $\mathcal{W}$ are at infinity in the tangent space. This motivates to regard

$$U^{(\gamma)} = \exp^{-1}_{\mathbb{1}_\mathcal{W}}(\epsilon M\delta_\gamma + (1-\epsilon)\mathbb{1}_\mathcal{W}) \in T_0\mathcal{W}, \qquad \gamma \in [c]^n \tag{5.49}$$

corresponding to *smoothed extremal points* of $\mathcal{W}$ for some small smoothing constant $0 < \epsilon \ll 1$. For every choice of ϵ, the sets (5.48) form a Voronoi tesselation of $T_0\mathcal{W}$ with anchor points (5.49).

5.3.3 Flow Matching

We aim to learn discrete data distributions by CNAF, employing either of the parameterizations (5.46) or (5.45). The payoff function $F_{\mathfrak{p}}$ will be parameterized as a deep neural network. To this end, likelihood maximization to optimize the bound (5.43) after dequantization can be used in principle. The most computationally expensive element of likelihood maximization is forward and adjoint (backward) simulation of assignment flow integral curves to evaluate the flow map of (5.7) and its gradient. In order to compute likelihood and its stochastic gradient at sufficient precision, tens of forward and backward passes through the payoff function network $F_{\mathfrak{p}}$ are typically required [43, 78]. To improve scalability, [125] propose *flow matching*, a simulation-free alternative to likelihood-based learning. If we had access to a *probability path* $t \mapsto \pi_t \in \mathcal{P}(T_0\mathcal{W})$ such that $\pi_0 = \mathcal{N}_0$ is a reference distribution and $\pi_1 = \pi$ represents the target distribution by $p = \mathbb{E}_{V\sim\pi}[T(\exp_{\mathbb{1}_\mathcal{W}}(V))]$ as well as a vector field $u\colon T_0\mathcal{W} \to T_0\mathcal{W}$ whose flow-map pushes forward $\mathcal{N}_0$ to π_t for every time $t \in [0,1]$, we could learn the payoff function of (5.7) by matching the flow generated by u. More concretely, returning to the tangent space parameterization (3.35) of assignment flows

$$\dot{V}(t) = \Pi_0 F_{\mathfrak{p}}(\exp_{\mathbb{1}_\mathcal{W}}(V)), \qquad V(0) = \exp^{-1}_{\mathbb{1}_\mathcal{W}}(W_0), \tag{5.50}$$

flow matching is a regression task which learns $\mathfrak{p}$ in (5.50) such that the vector field on the right hand side matches u at every $V \in T_0\mathcal{W}$.

Without prior knowledge, flow matching appears intractable, because we do not have access to a probability path or vector field satisfying these requirements. However, a core contribution of [125] is that flow matching can be performed by matching *conditional* vector fields for each data point. For every $\gamma \in [c]^n$, we can design a conditioned probability path $\pi_t(\cdot|\gamma)$ such that $\pi_0 = \mathcal{N}_0$ and $\pi_1(\cdot|\gamma) \in \mathcal{P}(T_0\mathcal{W})$ is a tractable distribution concentrated close to $U^{(\gamma)}$. Then the marginal probability path

$$\pi_t(V) = \sum_{\gamma\in[c]^n} p_\gamma \pi_t(V|\gamma) \tag{5.51}$$

satisfies $\pi_0 = \mathcal{N}_0$ and $\mathbb{E}_{V\sim\pi_1}[T(\exp_{\mathbb{1}_\mathcal{W}}(V))]$ closely approximates p. [125, Theorem 1] now shows that π_t in (5.51) is generated by the vector field

$$u_t(V) = \sum_{\gamma\in[c]^n} u_t(V|\gamma)\frac{\pi_t(V|\gamma)p_\gamma}{\pi_t(V)}dV \tag{5.52}$$

where $u_t(\cdot|\gamma)$ generates the conditional probability path $\pi_t(\cdot|\gamma)$. Strikingly, [125, Theorem 2] further shows that, if π_t has full support on $T_0\mathcal{W}$, the conditional flow matching objective

$$\widetilde{\mathcal{L}}_{\mathrm{CFM}}(\mathfrak{p}) = \mathbb{E}_{t\sim\mathcal{U}[0,1],\gamma\sim p, V\sim\pi(\cdot|\gamma)}\|\Pi_0 F_{\mathfrak{p}}(\exp_{\mathbb{1}_\mathcal{W}}(V)) - u_t(W|\gamma)\|^2 \tag{5.53}$$

has the same gradient with respect to $\mathfrak{p}$ as the intractable flow matching objective

$$\widetilde{\mathcal{L}}_{\mathrm{FM}}(\mathfrak{p}) = \mathbb{E}_{t\sim\mathcal{U}[0,1],V_0\sim\pi_0}\|\dot{\psi}_{\mathfrak{p}}(W_0,t) - u_t(\psi_{\mathfrak{p}}(W_0,t))\|^2. \tag{5.54}$$

We can further improve on (5.53) by employing a generalization of flow matching to Riemannian manifolds [42] directly on $\mathcal{W}$. To this end, conditional vector fields $u_t(\cdot|\gamma)\colon \mathcal{W} \to T_0\mathcal{W}$ are constructed such that their flow transports the reference distribution along geodesic curves of $\mathcal{W}$. Recall that e-geodesics on $\mathcal{W}$ are curves

$$W_t = \exp_{W_0}(tV), \qquad t \geq 0 \tag{5.55}$$

starting at an initial point W_0 in direction $V \in T_0\mathcal{W}$. For given $\gamma \in [c]^n$, we can define the conditional vector field $u_t(\cdot|\gamma)$ such that every initial point $W_0 \in \mathcal{W}$, governed by the reference distribution

$$\pi_0 = (\exp_{\mathbb{1}_\mathcal{W}})_\sharp \mathcal{N}_0 \in \mathcal{P}(\mathcal{W}) \tag{5.56}$$

is transported to $\exp_{\mathbb{1}_\mathcal{W}}(U^{(\gamma)})$ after time $t = 1$ along the e-geodesic curve W_t^γ uniquely defined by these requirements. This construction ensures that the pushforward measure $\pi_1 \in \mathcal{P}(\mathcal{W})$ generated by the marginal flow map closely approximates the target distribution through $p \approx \mathbb{E}_{W\sim\pi_1}[T(W)]$. With $V_0 = \exp_{\mathbb{1}_\mathcal{W}}^{-1}(W_0)$, the respective geodesic curve reads

$$W_t^\gamma = \exp_{W_0}(t(U^{(\gamma)} - V_0)) = \exp_{\mathbb{1}_\mathcal{W}}(V_0 + t(U^{(\gamma)} - V_0)) \tag{5.57}$$

and we find the sought conditional vector field along (5.57) by differentiation

$$u_t(W_t^\gamma|\gamma) = \mathcal{R}_{W_t^\gamma}[U^{(\gamma)} - V_0]. \tag{5.58}$$

Note that (5.58) still depends on the initial point W_0 through V_0. Further, (5.58) has the shape of an assignment flow on $\mathcal{W}$, which suggests to make a corresponding assignment flow ansatz (5.7) with parameterized payoff function $F_\mathfrak{p}$ for flow matching. By measuring discrepancy in the Fisher Rao metric on $T_0\mathcal{W}$, we find the *Riemannian conditional flow matching* (RCFM) objective

$$\begin{aligned}
\mathcal{L}_{\mathrm{RCFM}}(\mathfrak{p}) &= \mathbb{E}_{t\sim\mathcal{U}[0,1],\gamma\sim p,V_0\sim\mathcal{N}_0} \|\mathcal{R}_{W_t^\gamma}[F_\mathfrak{p}(W_t^\gamma)] - \mathcal{R}_{W_t^\gamma}[U^{(\gamma)} - V_0]\|_{W_t^\gamma}^2 && \text{(5.59a)}\\
&= \mathbb{E}_{t\sim\mathcal{U}[0,1],\gamma\sim p,V_0\sim\mathcal{N}_0} \|\mathcal{R}_{W_t^\gamma}[F_\mathfrak{p}(W_t^\gamma) - (U^{(\gamma)} - V_0)]\|_{W_t^\gamma}^2 && \text{(5.59b)}
\end{aligned}$$

inkeeping with the construction of [42].

Learning assignment flows by minimizing (5.59) is efficient and scalable, because no simulation of integral curves is required. It also allows for much flexibility in choosing a reference distribution, because only samples from $\mathcal{N}_0$ are required for training.

5.3.4 Likelihood Evaluation

After learning a payoff function $F_\mathfrak{p}$ through minimization of (5.59), we are able to represent the target distribution through either of the approaches (5.46) or (5.45) without retraining. In particular, data likelihood is not available during training and differs between both approaches. Since in both cases, the learned model is built on a continuous normalizing flow, we can leverage the methods described in Section 5.3.1 to develop methods for likelihood computation.

In the first case, the probability of $\gamma \in [c]^n$ under the learned model is

$$q_{\mathfrak{p}}(\gamma) = \mathbb{E}_{W\sim\nu_{\mathfrak{p}}}[T(W)_\gamma]. \tag{5.60}$$

An unbiased estimator for this quantity is easily found by replacing the expectation with a mean over samples. However, such an estimate is subject to a *rare events* problem, like the one described in Section 5.3.2. $\nu_{\mathfrak{p}}$ concentrates close to extremal points of $\mathcal{W}$ and thus, estimating (5.60) by a mean over samples is plagued by most samples yielding essentially vanishing probability $T(W)_\gamma$. In order to build a more effective estimator, we thus employ an *importance sampling* approach. This builds on the idea that samples are most relevant if they lie in A_γ. For fixed γ of interest, construct a proposal distribution $\zeta \in \mathcal{P}(\mathcal{W})$ which has full support but is concentrated on A_γ. Details of such a construction are discussed in Appendix B. We now re-write (5.60) as

$$q_{\mathfrak{p}}(\gamma) = \mathbb{E}_{W\sim\nu_{\mathfrak{p}}}[T(W)_\gamma] = \int \nu_{\mathfrak{p}}(W)T(W)_\gamma dW \tag{5.61a}$$

$$= \int \zeta(W)\frac{\nu_{\mathfrak{p}}(W)T(W)_\gamma}{\zeta(W)}dW \tag{5.61b}$$

$$= \mathbb{E}_{W\sim\zeta}\Big[\frac{\nu_{\mathfrak{p}}(W)T(W)_\gamma}{\zeta(W)}\Big]. \tag{5.61c}$$

Replacing the last expectation by a mean over *importance samples* drawn from ζ yields an effective estimator. In practice, (5.61) is still prone to numerical underflow in high dimensions. To remedy this, we focus on the log-likelihood instead. A simple aproach is using Jensen's inequality which gives

$$\log q_{\mathfrak{p}}(\gamma) = \log \mathbb{E}_{W\sim\zeta}\Big[\frac{\nu_{\mathfrak{p}}(W)T(W)_\gamma}{\zeta(W)}\Big] \geq \mathbb{E}_{W\sim\zeta}[\log \nu_{\mathfrak{p}}(W) + \log T(W)_\gamma - \log \zeta(W)]. \tag{5.62}$$

It turns out that, for the relatively narrow purpose of avoiding numerical underflow, we can improve on (5.61), avoiding the Jensen gap. To this end, consider importance samples $\{W_k\}_{k\in[m]}$ drawn from ζ. Then the sought estimator of log-likelihood reads

$$\log q_{\mathfrak{p}}(\gamma) \approx \log \frac{1}{m}\sum_{k\in[m]} \frac{\nu_{\mathfrak{p}}(W_k)T(W_k)_\gamma}{\zeta(W_k)} \tag{5.63a}$$

$$= -\log m + \log \sum_{k\in[m]} \exp\Big(\log \frac{\nu_{\mathfrak{p}}(W_k)T(W_k)_\gamma}{\zeta(W_k)}\Big) \tag{5.63b}$$

$$= -\log m + \log \sum_{k\in[m]} \exp\Big(\log \nu_{\mathfrak{p}}(W_k) + \log T(W_k)_\gamma - \log \zeta(W_k)\Big). \tag{5.63c}$$

This simple trick allows stable numerical evaluation of the estimator (5.61) by leveraging stabilized implementation of the logsumexp function. Note that

$$\log T(W)_\gamma = \log \prod_{i\in[n]} W_{i,\gamma_i} = \sum_{i\in[n]} \log W_{i,\gamma_i} \tag{5.64}$$

which is not prone to underflow.

For the second approach (5.45), likelihood under the model reads

$$q^{\mathrm{r}}_{\mathfrak{p}}(\gamma) = \mathbb{P}_{W\sim\nu_{\mathfrak{p}}}(W \in A_\gamma) = \mathbb{E}_{W\sim\nu_{\mathfrak{p}}}(\mathbb{1}_{A_\gamma}(W)) \tag{5.65}$$

and we can analogously use the estimator

$$\log q^{\mathrm{r}}_{\mathfrak{p}}(\gamma) \approx -\log m + \log \sum_{k\in[m]} \exp\Big(\log \nu_{\mathfrak{p}}(W_k) + \log \mathbb{1}_{A_\gamma}(W_k) - \log \zeta(W_k)\Big) \tag{5.66}$$

for importance samples $\{W_k\}_{k\in[m]}$. Note the conventions $\log 0 = -\infty$ and $\exp(-\infty) = 0$ in (5.66).

In both cases, the log-likelihood of importance samples under the learned pushforward distribution $\nu_{\mathfrak{p}}$ is required. If log-likelihood under the reference distribution $\mathcal{N}_0$ is tractable, this quantity can be approximated by using the methods established in Section 5.3.1. Specifically, the instantaneous change of variables formula of Theorem 5.3 represents likelihood under the pushforward distribution $\nu_{\mathfrak{p}}$ by likelihood under $\mathcal{N}_0$ and a correction for non-vanishing divergence of the transport vector field (5.35). The latter can be approximated by combining adjoint integration with Hutchinson's trace estimator (Lemma 5.4).

5.3.5 Experiments

Simple Discrete Distributions We learn three simple distributions, each a joint distribution of two coupled random variables. Histograms of samples from the learned distributions are shown in Figure 5.5. This experiment serves to illustrate that the approach (5.6) is able to represent coupled random variables, even though every $W \sim \nu_{\mathfrak{p}}$ only represents a factorizing distribution $T(W)$ of uncoupled variables. The simplest example is the joint distribution of two strongly coupled binary random variables shown as blue dot in Figure 5.2. Our fit to this target distribution is represented by the right plot in Figure 5.5, showing the (empirical) frequencies of all $c^n = 4$ class configurations. For this task, we choose $F_{\mathfrak{p}}$ as a linear function (of vectorized state) as in (3.63) for a densely connected graph. The left two images show results of fitting target distributions on $c^n = 91^2 = 8281$ class configurations. Here, we choose $F_{\mathfrak{p}}$ as a three-layer neural network with dense linearities, hidden dimensions $(256, 256)$ and ReLU activation.

Generating Image Segmentations In image segmentation, a joint assignment of classes to pixels is usually sought conditioned on the pixel values themselves. Here, we instead focus on the *unconditional* discrete distribution of segmentations, without regard to the original pixel data. These discrete distributions are very high-dimensional in general, with $N = c^n$ increasing exponentially in the number of pixels. We perform Riemannian conditional flow matching on $\mathcal{W}$ to learn assignment flows (5.7) which approximate this discrete distribution via (5.6) or (5.45). To this end, we parametrize F_θ by the UNet architecture of [56] (details in Appendix D.1) and train on the segmentations of Cityscapes [49], downsampled to $c = 8$ classes and resolution 128×256, as well as MNIST [118], regarded as binary $c = 2$ segmentations of 28×28 pixel images. We pad binarized MNIST

Figure 5.5: Histogram of samples from our model fitting the joint distribution of $n = 2$ discrete random variables. *Left and middle*: $c = 91$ classes per variable. *Right*: $c = 2$ classes per variable. All three plots show joint distribution probabilities. Clearly, the model is able to fit multi-modal joint distributions which do not factorize into independent marginals. The plot on the right is the joint distribution shown as blue dot in Figure 5.2.

data to size 32×32, in order to make it compatible with spatial downsampling employed by the chosen UNet architecture. Figure 5.6 shows samples from the learned distribution of Cityscapes segmentations randomly drawn from our model by rounding each sample $W \sim \nu_{\mathfrak{p}}$ to the nearest extremal point. Figure 5.7 illustrates the difference between samples from $T(W)$ as in (5.6) and rounding W to the nearest extremal point as in (5.45) for our learned distribution of binarized MNIST digits. During training of the model, we use $\epsilon = 0.01$ in (5.49) for smoothing of training data. Suppose by drawing a sample $W \sim \nu_{\mathfrak{p}}$ from the learned model, we match a smoothed extremal point $W = \exp_{\mathbb{1}_{\mathcal{W}}}(U_\gamma)$, $\gamma \in [c]^n$ exactly. For MNIST, $c = 2$ classes are available for each node. When drawing $\widehat{\gamma} \sim T(W)$, the probability of $\widehat{\gamma}_i \neq \gamma_i$ for any given $i \in [n]$ due to smoothing (5.49) is $\epsilon - \epsilon/c = 0.005$. Based on this rationale, it is to be expected that on average one in 200 nodes is randomly assigned a different class from the one found by rounding. Accordingly, the binarized MNIST samples in Figure 5.7 are expected to contain around $32^2/200 = 5.12$ randomly flipped pixels in the second row, compared to the first.

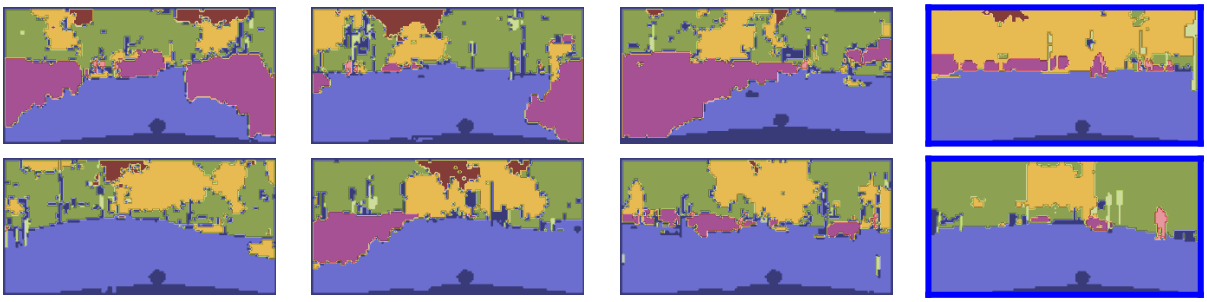

Figure 5.6: *Left:* Random samples drawn from our model trained on discrete Cityscapes segmentation data ($c = 8$ classes) at resolution 128×256. *Right with blue border:* Randomly drawn training data.

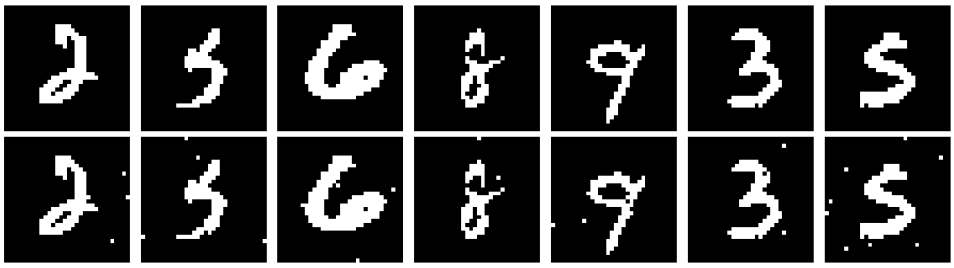

Figure 5.7: Samples from our model of the binarized MNIST data distribution, learned via RCFM (5.59). *First row:* $W \sim \nu_{\mathfrak{p}}$ rounded to the nearest extremal point as in (5.45). *Second row:* samples from $T(W)$ as in (5.46). The number of randomly flipped labels is dependent on the smoothing (5.49) employed in training ($\epsilon = 0.01$). Details of architecture and training procedure are listed in Appendix D.1.

Numerical Likelihood Evaluation In general, sampling integrals over high-dimensional domains is a difficult task. We evaluate the effectiveness of importance sampling approaches of Section 5.3.4 empirically by plotting the relative error of likelihoods computed from varying number of importance samples. We perform this experiment on our learned model of binarized MNIST, making the domain of integration $(c-1)n = 1024$ dimensional. The reference value for the integral is computed from 10^4 samples. Mean and standard deviation are evaluated over the first 50 data from the MNIST test set. Based on the evaluation shown in Figure 5.8, we conclude that on the order of 10^2 to 10^3 samples are required to achieve close likelihood approximation for a single datum.

We further compute the average likelihood of all 10^4 test data. To reduce computation, we use only 100 samples per datum, expecting that inaccuracy in likelihood evaluation for single data averages out in the mean. This assumption is based on the observed variance in Figure 5.8, independence of test data and the rationale that (5.66) merely numerically stabilizes the underlying unbiased estimator of (5.65). Further, as is the predominant practice in prior work [78, 41], we only use a single sample when employing Hutchinson's trace estimator (Lemma 5.4). The empirical distribution of likelihood for MNIST test data is shown in Figure 5.9. The empirical mean and standard deviation are 1.70 ± 0.05 Bits per dimension.

By subjective comparison of the sample quality in Figure 5.7 our testset likelihood appears low (high Bits per dimension). For comparison, the closely related continuous normalizing flow of [78] achieves 0.99 Bits per dimension on MNIST. Our method differs from [78] in a number of ways, including architecture choice and binarization of MNIST data. We ascribe the discrepancy between likelihoods primarily to the difference in training objective. [78] train by likelihood maximization, which serves to minimize relative entropy to the data distribution (5.38). Our flow matching approach by comparison does not specifically optimize data likelihood. It is thus to be expected that our model does not achieve the same likelihood and, by extension, does not fit the data as well in terms of relative entropy, as likelihood-based methods. However, good sample quality in Figure 5.7 indicates that our model has good fit to the data distribution, it merely does not optimize for relative entropy in particular. Evaluation of generative models in terms of sample

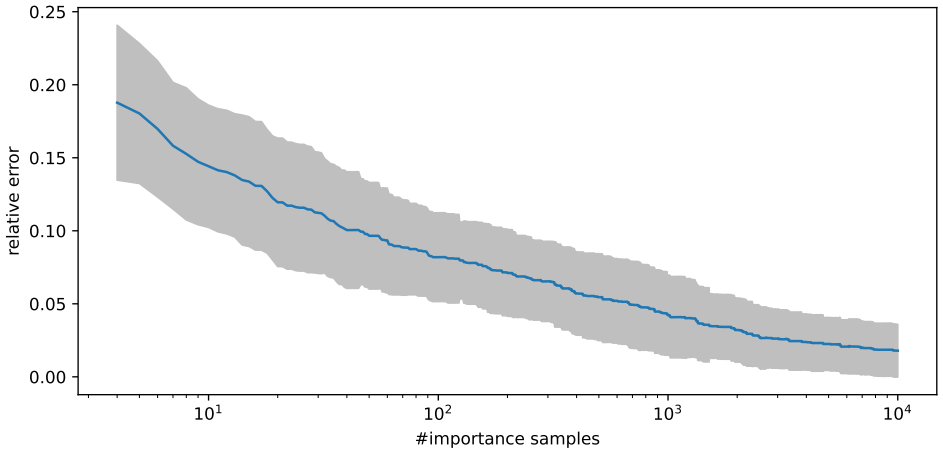

Figure 5.8: Convergence of importance sampling the integral (5.65) for our generative model of MNIST data. Mean and standard deviation are evaluated over the first 50 data from the MNIST test set.

quality and likelihood has been studied by [201]. The authors conclude that both metrics can be seen as independent in practice, demonstrating that models can simultaneously achieve high likelihood and poor sample quality.

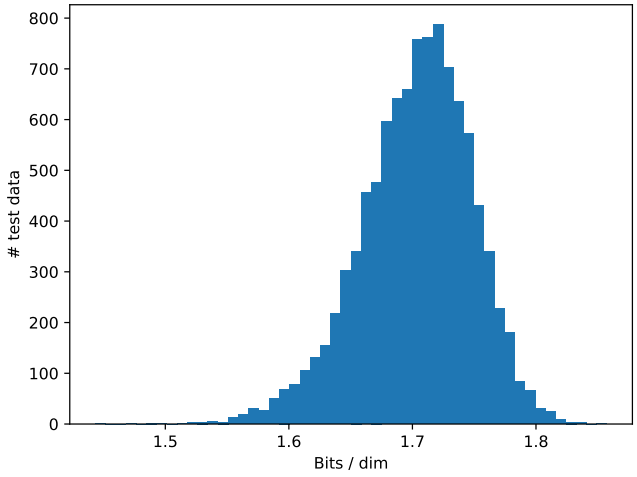

Figure 5.9: Empirical distribution of likelihood (Bits/dim) of MNIST test data. The mean value is 1.70 Bits/dim, standard deviation is 0.05 Bits/dim.

6 Certified Classification

In this second part of the thesis, we work toward the development of deep learning methods for structured prediction which are rooted in statistical learning theory. In Chapter 7, our discussion will culminate in the development of a new PAC-Bayesian risk certificate for structured prediction. Here, we start with a simpler classification method and associated risk certificate, illustrating current PAC-Bayesian methodology in the process and developing some new methods as well. In particular, we make the following contributions.

1. We construct a specific PAC-Bayesian hypothesis class of classifiers, built on a partially-randomized architecture which transforms all stochastic parameters affine-linearly.

2. We investigate different strategies for computing bounds on the expected empirical risk that are needed for numerical evaluation of PAC-Bayesian risk certificates and construct a particularly efficient one for our hypothesis class.

Recall the PAC-Bayesian construction presented in Section 2.2. We assume to have access to a sample

$$Z = ((X_1, Y_1), \ldots, (X_m, Y_m)) \sim \mu^m. \tag{6.1}$$

of $m > 0$ i.i.d. data drawn from an unknown distribution $\mu \in \mathcal{P}(\mathcal{X} \times \mathcal{Y})$. Here, we focus on image classification, so $\mathcal{X}$ is a vector space of images and $\mathcal{Y} = [c]$ is the discrete set of $c > 0$ classes. Working within the statistical learning framework of *uniform convergence*, the first step is to define a hypothesis class $\mathcal{H}$ of functions $\phi_{\mathfrak{p}} \colon \mathcal{X} \to \mathcal{Y}$, each parameterized by a unique parameter vector $\mathfrak{p}$. We assume the space of these parameters is $\mathbb{R}^d$ and identify $\mathcal{H}$ with it. Following the PAC-Bayesian construction, we will strategically construct $\mathcal{H}$ as well as distributions $\pi \in \overline{\mathcal{P}}(\mathcal{H})$ and $\rho \in \overline{\mathcal{P}}(\mathcal{H})$, respectively called PAC-Bayes *prior* and PAC-Bayes *posterior* to build a *self-certified* classification method. This means that, in contrast to the established approach of evaluating a model's ability to generalize on held-out test data, our PAC-Bayesian construction simultaneously learns a (stochastic) classifier and certifies its risk on unseen data from the same distribution. The certificate comes in the form of a high-probability upper bound on expected model risk under the posterior.

6.1 A PAC-Bayesian Classifier

Key to our approach is to restrict π and ρ to a subset of distributions which leave *most components of the parameter vector* $\mathfrak{p}$ *deterministic*. This is similar to the idea of a Bayesian last layer [72, 114] operating on features extracted by a deterministic map. Assuming again the PAC-Bayesian viewpoint, we formalize this idea by partitioning parameter vectors into a deterministic and a stochastic part

$$\mathfrak{p} = (\mathfrak{p}^{\mathsf{d}}, \mathfrak{p}^{\mathsf{s}}) \in \mathbb{R}^{d_0} \times \mathbb{R}^{d_1}, \qquad d_0 + d_1 = d \tag{6.2}$$

and structuring π and ρ according to

$$\pi = \delta_{\mathfrak{p}^{\mathsf{d}}} \times \pi^{\mathsf{s}} \qquad \rho = \delta_{\mathfrak{p}^{\mathsf{d}}} \times \rho^{\mathsf{s}} \qquad \pi^{\mathsf{s}}, \rho^{\mathsf{s}} \in \overline{\mathcal{P}}(\mathbb{R}^{d_1}). \tag{6.3}$$

With regard to PAC-Bayesian risk certification, the only restriction on prior and posterior, besides the way they are informed by data, is absolute continuity of ρ with respect to π. This ensures that their relative entropy is well-defined.

Lemma 6.1 (Relative Entropy of Prior and Posterior) *Fix a vector of deterministic parameters* $\mathfrak{p}^{\mathsf{d}} \in \mathbb{R}^{d_0}$ *and let* π^{s} *and* ρ^{s} *be equivalent measures, i.e.* $\pi^{\mathsf{s}} \ll \rho^{\mathsf{s}}$ *and* $\rho^{\mathsf{s}} \ll \pi^{\mathsf{s}}$. *Then* $\rho = \delta_{\mathfrak{p}^{\mathsf{d}}} \times \rho^{\mathsf{s}}$ *is absolutely continuous with respect to* $\pi = \delta_{\mathfrak{p}^{\mathsf{d}}} \times \pi^{\mathsf{s}}$ *(*$\rho \ll \pi$*) and*

$$\mathrm{KL}(\rho, \pi) = \mathrm{KL}(\rho^{\mathsf{s}}, \pi^{\mathsf{s}}). \tag{6.4}$$

Proof. Let A be a measurable subset of $\mathcal{H}$ with $\pi(A) = 0$ and denote by A_0 and A_1 the projection of A onto the respective coordinates in the partition (6.2). Then

$$\delta_{\mathfrak{p}^{\mathsf{d}}}(A_0)\pi^{\mathsf{s}}(A_1) = 0 \tag{6.5}$$

and thus, at least one of the factors needs to vanish. If the first vanishes, this directly implies $\rho(A) = 0$. In addition, $\pi^{\mathsf{s}}(A_1) = 0$ implies $\rho^{\mathsf{s}}(A_1) = 0$ and thus $\rho(A) = 0$ because π^{s} and ρ^{s} are equivalent measures. It follows $\rho \ll \pi$. Because the factors in ρ resp. π are independent, relative entropy decomposes as

$$\mathrm{KL}(\rho, \pi) = \mathrm{KL}(\rho^{\mathsf{s}}, \pi^{\mathsf{s}}) + \underbrace{\mathrm{KL}(\delta_{\mathfrak{p}^{\mathsf{d}}}, \delta_{\mathfrak{p}^{\mathsf{d}}})}_{=0} \tag{6.6}$$

□

In particular, Lemma 6.1 shows that the chosen structure (6.3) of π and ρ is not a barrier to PAC-Bayesian risk certification if both distributions of stochastic parameters π^{s} and ρ^{s} have full support and deterministic parameters $\mathfrak{p}^{\mathsf{d}}$ are shared between prior and posterior.

The second core idea behind our approach is to *transform stochastic parameters only affine-linearly*. This ensures that normal distribution of stochastic parameters generates a normal distribution of class predictions for each input datum. More precisely, suppose every $\phi_{\mathfrak{p}} \in \mathcal{H}$ is composed of $\widetilde{\phi}_{\mathfrak{p}} \colon \mathcal{X} \to T_0\mathcal{S}_c$ and rounding

$$\phi_{\mathfrak{p}}(x) = \mathrm{argmax}_{j \in [c]}\, \widetilde{\phi}_{\mathfrak{p}}(x)_j. \tag{6.7}$$

The entries of $\widetilde{\phi}_{\mathfrak{p}}(x) \in T_0\mathcal{S}_c$ are called *classification logits*. If $\widetilde{\phi}_{\mathfrak{p}}(x)$ defines an affine transformation of stochastic parameters for each $x \in \mathcal{X}$ and $\mathfrak{p}^{\mathfrak{s}}$ is a random variable with normal distribution, then the classification logits also follow a normal distribution on $T_0\mathcal{S}_c$ for each x.

6.1.1 Linearized Assignment Flows

We now describe a particular choice of data-dependent affine transformation which is based on the linearization of an assignment flow. The core idea is that linear ODEs transform their initialization affine-linearly for fixed end time. In this section, we assume a fixed set of parameters, which is later randomized in Section 6.1.2. Initially, assume general EGN dynamics (3.63) in vectorized form, which read

$$\dot{w}(t) = \mathcal{R}^{\mathfrak{v}}_{w(t)}[\overline{A}w(t)], \qquad w(0) = w_0 = \mathrm{vec}_r(W_0). \tag{6.8}$$

By Theorem 3.3, the dynamics (6.8) can be parameterized on the tangent space at $w(0)$ as

$$w(t) = \exp^{\mathfrak{v}}_{w_0}(v(t)) \tag{6.9a}$$
$$\dot{v}(t) = \Pi^{\mathfrak{v}}_0 \overline{A} \exp^{\mathfrak{v}}_{w_0}(v(t)), \qquad v(0) = 0. \tag{6.9b}$$

We now linearize the vector field in (6.9b).

Proposition 6.2 (Linearized Assignment Flow) *The system of equations*

$$w(t) = \exp^{\mathfrak{v}}_{w_0}(v(t)), \tag{6.10a}$$
$$\dot{v}(t) = \Pi^{\mathfrak{v}}_0 \overline{A}(w_0 + \mathcal{R}^{\mathfrak{v}}_{w_0} v(t)), \qquad v(0) = 0 \tag{6.10b}$$

approximates (6.8). *The initial value problem in* (6.10) *has the closed-form solution*

$$v(t) = t\varphi(\mathsf{M})v_D, \qquad \mathsf{M} = t\Pi^{\mathfrak{v}}_0 \overline{A} \mathcal{R}^{\mathfrak{v}}_{w_0}, \quad v_D = \Pi^{\mathfrak{v}}_0 \overline{A} w_0, \tag{6.11}$$

where $\varphi\colon \mathbb{R}^{nc\times nc} \to \mathbb{R}^{nc\times nc}$ *denotes the analytical function*

$$\varphi(z) = \frac{e^z - 1}{z} \tag{6.12}$$

with matrix argument.

Proof. Due to (2.46), the vectorized lifting map has the differential

$$d\exp^{\mathfrak{v}}_{w_0}(v)[u] = \mathcal{R}^{\mathfrak{v}}_{w_0} u \tag{6.13}$$

which we use to compute the desired linearization

$$\dot{v}(t) \approx \Pi^{\mathfrak{v}}_0 \overline{A}\Big(\exp^{\mathfrak{v}}_{w_0}(v_0) + d\exp^{\mathfrak{v}}_{w_0}(v_0)[v(t) - v_0]\Big) = \Pi^{\mathfrak{v}}_0 \overline{A}(w_0 + \mathcal{R}^{\mathfrak{v}}_{w_0} v(t)). \tag{6.14}$$

Since (6.10b) is linear in v, Duhamel's variation of constants formula [200] yields (6.11). □

We call (6.10) *linearized assignment flow* (LAF), because it is the result of linearizing (6.8). Note however, that the system (6.10) defines a nonlinear transformation of w_0, which is parameterized by the linear ODE (6.10b). In addition, LAF classifiers are nonlinear models due to the dependence of $A = \Pi_0^{\mathfrak{v}} \overline{A} \mathcal{R}_{w_0}^{\mathfrak{v}}$ on w_0. The term *LAF* has been used in the literature to refer to similar linearizations of (other) assignment flows [225, 226]. In these works, the observation is put forward that linearizing in the described manner largely preserves the qualitative behavior of assignment flows in practice. In turn, such linearizations are easier to study and treat numerically. Their solution can be efficiently approximated by using Krylov subspace methods and speciallized approaches have been developed to approximate their gradient with respect to parameters [226]. Note that the method of risk certification proposed in the following sections works with the linearized system (6.10) instead of the motivating system (6.8). Thus, the constructed bounds hold irrespective of how closely (6.10) approximates (6.8).

We briefly describe the concept of Krylov subspace methods to evaluate (6.11). For $0 < k \leq nc$, the linear subspace

$$\mathcal{K}_k = \operatorname{span}\{v_D, \mathsf{M} v_D, \mathsf{M}^2 v_D, \ldots, \mathsf{M}^{k-1} v_D\} \subseteq \mathbb{R}^{nc} \tag{6.15}$$

is called *Krylov subspace* of order k generated by M and v_D. For simplicity, we assume that $\mathcal{K}_k$ has dimension k. Arnoldi's method [9] (Algorithm 1) employs interleaved power

Algorithm 1: Arnoldi's method

Data: $v_D \neq 0 \in \mathbb{R}^m$, $\mathsf{M} \in \mathbb{R}^{nc \times nc}$, $nc > 0$, $0 < k \leq nc$
Result: Orthonormal basis $\mathsf{Q} \in \mathbb{R}^{nc \times k}$ of the Krylov subspace (6.15) and upper Hessenberg matrix $\mathsf{H}_k \in \mathbb{R}^{k \times k}$ satisfying (6.16)

```
v_1 ← v_D/‖v_D‖;
for j = 1, ..., k do
    w ← M v_j;
    for i = 1, ..., j do
        h_ij ← ⟨w, v_i⟩;
        w ← w − h_ij v_i;
    end
    h_{j+1,j} ← ‖w‖_2;
    v_{j+1} ← w/h_{(j+1),j};
end
H_k ← entries h_ij and Q ← columns v_j for i, j ∈ [k];
```

iteration and orthogonalization to construct an orthonormal basis $\mathsf{Q} \in \mathbb{R}^{nc \times k}$ of $\mathcal{K}_k$ as well as an upper Hessenberg matrix $\mathsf{H}_k = \mathsf{Q}^\top \mathsf{M} \mathsf{Q} \in \mathbb{R}^{k \times k}$ such that the *Arnoldi relations*

$$\mathsf{M}\mathsf{Q} = \mathsf{Q}\mathsf{H}_k + r_k e_k^\top \tag{6.16a}$$

$$\mathsf{Q}^\top \mathsf{M} \mathsf{Q} = \mathsf{H}_k \tag{6.16b}$$

hold for $r_k = h_{k+1,k} v_{k+1}$ defined in the last step of Algorithm 1. The power iteration and orthogonalization can also be performed separately, which we show in Appendix C.3. This

does not in itself improve numerical stability or reduce computational effort. However, it can help to simplify practical implementation of Arnoldi's method, because efficient and numerically stable QR decomposition for the purpose of orthogonalization is readily available in many numerical libraries.

Motivated by (6.16), [70] propose the following approximation[1]

$$t\varphi(\mathsf{M})v_D \approx t\varphi(\mathsf{Q}\mathsf{H}_k\mathsf{Q}^\top)v_D = t\mathsf{Q}\varphi(\mathsf{H}_k)\mathsf{Q}^\top v_D \tag{6.17}$$

where the second equality is clear from the series expansion

$$\varphi(\mathsf{Q}\mathsf{H}_k\mathsf{Q}^\top) = \sum_{j=0}^{\infty} \frac{1}{(j+1)!}(\mathsf{Q}\mathsf{H}_k\mathsf{Q}^\top)^j = \sum_{j=0}^{\infty} \frac{1}{(j+1)!}\mathsf{Q}\mathsf{H}_k^j\mathsf{Q}^\top \tag{6.18}$$

because $\mathsf{Q}^\top\mathsf{Q} = \mathbb{I}_k$. In practice, small $k \ll nc$ suffices to achieve close approximation in (6.17) and $\varphi(\mathsf{H}_k)$ can be evaluated using standard methods like eigendecomposition. For a detailed analysis of evaluating matrix functions using Krylov subspaces, we refer to [174]. An important detail of the above Krylov approximation is that $\mathcal{K}_k$ is generated by v_D, defined in (6.11) and $\varphi(\mathsf{M})$ acts on the same vector v_D in (6.17). This is crucial to achieve close approximation. In particular, (6.17) only effectively approximates the action of $\varphi(\mathsf{M})$ on *vectors*. The task of evaluating the action on a matrix needs to be broken down into separate approximation of the action on each column.

6.1.2 Randomization

We will approach the construction of a randomized classifier by randomizing the initialization v_0 in (6.10b) instead of fixing it to zero. This is conceptually similar to the case considered in [76] where the authors study pushforward distributions of linearized assignment flows under uncertain initialization. For arbitrary initialization $v_0 \in T_0\mathcal{W}$, the linear ODE

$$\dot{v}(t) = \Pi_0^{\mathfrak{v}}\overline{A}(w_0 + \mathcal{R}_{w_0}^{\mathfrak{v}}v(t)) = \mathsf{M}v(t) + v_D, \qquad v(0) = v_0 \tag{6.19}$$

has the closed-form solution

$$v(t) = t\varphi(\mathsf{M})v_D + \operatorname{expm}(\mathsf{M})v_0, \tag{6.20}$$

extending (6.11) by a second summand which is linear in v_0. We will define a notion of normal distribution for v_0 which is supported on $T_0\mathcal{W}$. This has controlled behavior under affine transformations, but does not have a density with respect to the Lebesgue measure on $\mathbb{R}^{n\times c}$.

Definition 6.3 (Normal Distribution on $T_0\mathcal{W}$) *Let $\mu \in T_0\mathcal{W}$ and let $\mathcal{V} \in \mathbb{R}^{nc\times n(c-1)}$ be a matrix such that $\operatorname{vec}^{-1}(V_i) \in T_0\mathcal{W}$ for all column vectors V_i, $i \in [n(c-1)]$ of $\mathcal{V}$. Denote the componentwise standard normal distribution on $\mathbb{R}^{n(c-1)}$ by $\mathcal{N}_0^{n(c-1)}$. Then we call*

$$\mathcal{N}(\mu, \mathcal{V}\mathcal{V}^\top) = (\operatorname{vec}^{-1} \circ \mathcal{V})_\sharp \mathcal{N}_0^{n(c-1)} \tag{6.21}$$

a normal distribution *on $T_0\mathcal{W}$ with mean μ and (singular) covariance $\mathcal{V}\mathcal{V}^\top$.*

[1] The authors of [70] use the approximation (6.17) for the matrix exponential instead of φ, but their argument applies analogously to other entire matrix functions.

For the special case $n = 1$, this also defines a notion of normal distribution on $T_0\mathcal{S}_c$. Definition 6.3 clarifies the formal notation $\mathcal{N}(\mu, \mathcal{V}\mathcal{V}^\top)$ in ambient coordinates, where the covariance $\mathcal{V}\mathcal{V}^\top$ is a singular matrix. The relative entropy between two normal distributions on $T_0\mathcal{W}$ has a closed form which we compute next. Let $\widetilde{P} \in \mathbb{R}^{c\times(c-1)}$ be a matrix whose columns are a basis of $T_0\mathcal{S}_c$ and let $\widetilde{P}^\dagger$ be a pseudoinverse matrix in the sense that $\widetilde{P}^\dagger\widetilde{P} = \mathbb{I}_{c-1}$ and $\widetilde{P}\widetilde{P}^\dagger v = v$ for every $v \in T_0\mathcal{S}_c$. For example, using the m-coordinate basis, one can construct

$$\widetilde{P} = \begin{bmatrix} \mathbb{I}_{c-1} \\ -\mathbb{1}_{c-1}^\top \end{bmatrix} \in \mathbb{R}^{c\times(c-1)}, \qquad \widetilde{P}^\dagger = \begin{bmatrix} \mathbb{I}_{c-1} & 0 \end{bmatrix} \in \mathbb{R}^{(c-1)\times c} \tag{6.22}$$

which satisfy these constraints. Further, if we orthonormalize the columns of $\widetilde{P}$ and let $\widetilde{P}^\dagger = \widetilde{P}^\top$, then $\widetilde{P}^\top\widetilde{P} = \mathbb{I}_{c-1}$ is satisfied by orthonormality of columns and $\widetilde{P}\widetilde{P}^\top$ is the orthogonal projection operator onto $T_0\mathcal{S}_c$, which implies $\widetilde{P}\widetilde{P}^\top v = v$. Let

$$\mathfrak{P}\colon \mathbb{R}^{n(c-1)} \to T_0\mathcal{W}, \qquad \mathfrak{P}^\dagger\colon T_0\mathcal{W} \to \mathbb{R}^{n(c-1)} \tag{6.23}$$

be the linear operators which apply $\widetilde{P}$ resp. $\widetilde{P}^\dagger$ node-wise. With abuse of notation, we also use the symbol $\mathfrak{P}^\dagger$ to denote the respective operator with vectorized argument.

Lemma 6.4 (Relative Entropy of Normal Distributions on $T_0\mathcal{W}$) *Let $p_1 = \mathcal{N}(\mu_1, \mathcal{V}_1\mathcal{V}_1^\top)$ and $p_2 = \mathcal{N}(\mu_2, \mathcal{V}_2\mathcal{V}_2^\top)$ be normal distributions on $T_0\mathcal{W}$ in the sense of Definition 6.3 with full support on $T_0\mathcal{W}$. Define the multivariate normal distributions*

$$\widetilde{p}_i = \mathcal{N}(\mathfrak{P}^\dagger\mu_i, \Sigma_i), \qquad \Sigma_i = (\mathfrak{P}^\dagger\mathcal{V}_i)(\mathfrak{P}^\dagger\mathcal{V}_i)^\top, \qquad i \in \{1, 2\} \tag{6.24}$$

on $\mathbb{R}^{n(c-1)}$. Then it holds $\mathrm{KL}(p_1, p_2) = \mathrm{KL}(\widetilde{p}_1, \widetilde{p}_2)$.

Proof. Because $\widetilde{P}\widetilde{P}^\dagger v = v$ for every $v \in T_0\mathcal{S}_c$ and the columns of $\mathcal{V}_i$ are (vectorized) tangent vectors in $T_0\mathcal{W}$, it holds $\mathfrak{P}\mathfrak{P}^\dagger\mathcal{V}_i = \mathcal{V}_i$. Thus, we can write

$$p_i = \mathfrak{P}_\sharp\mathcal{N}(\mathfrak{P}^\dagger\mu_i, \Sigma_i) = \mathfrak{P}_\sharp\widetilde{p}_i, \qquad i \in \{1, 2\} \tag{6.25}$$

and, since p_i was assumed to have full support on $T_0\mathcal{W}$, $\widetilde{p}_i$ is a multivariate normal distribution on $\mathbb{R}^{n(c-1)}$ with full-rank covariance matrix. Denote the density of $\widetilde{p}_i$ with respect to the Lebesgue measure on $\mathbb{R}^{n(c-1)}$ by $\zeta_i\colon \mathbb{R}^{n(c-1)} \to \mathbb{R}$. Let $A \subseteq T_0\mathcal{W}$ be a measurable set and $\widetilde{A} \subseteq \mathbb{R}^{n(c-1)}$ its preimage under $\mathfrak{P}$. Then, due to (6.25), it holds $p_i(A) = \widetilde{p}_i(\widetilde{A})$. Further,

$$\int_A \frac{\zeta_1(\mathfrak{P}^\dagger x)}{\zeta_2(\mathfrak{P}^\dagger x)}\mathrm{d}p_2(x) = \int_{\widetilde{A}} \frac{\zeta_1(\mathfrak{P}^\dagger\mathfrak{P}y)}{\zeta_2(\mathfrak{P}^\dagger\mathfrak{P}y)}\mathrm{d}\widetilde{p}_2(y) = \int_{\widetilde{A}} \frac{\zeta_1(y)}{\zeta_2(y)}\zeta_2(y)\mathrm{d}y \tag{6.26a}$$

$$= \int_{\widetilde{A}} \zeta_1(y)\mathrm{d}y = \widetilde{p}_1(\widetilde{A}) = p_1(A) \tag{6.26b}$$

clarifies the shape of the Radon-Nikodým derivative

$$\frac{\mathrm{d}p_1}{\mathrm{d}p_2}(x) = \frac{\zeta_1(\mathfrak{P}^\dagger x)}{\zeta_2(\mathfrak{P}^\dagger x)}. \tag{6.27}$$

Using (6.27), we directly compute

$$\mathrm{KL}(p_1, p_2) = \int_{T_0\mathcal{W}} \log \frac{\mathrm{d}p_1}{\mathrm{d}p_2}(x)\mathrm{d}p_1(x) = \int_{T_0\mathcal{W}} \log \frac{\zeta_1(\mathfrak{P}^\dagger x)}{\zeta_2(\mathfrak{P}^\dagger x)}\mathrm{d}p_1(x) \tag{6.28a}$$

$$= \int_{\mathbb{R}^{n(c-1)}} \log \frac{\zeta_1(\mathfrak{P}^\dagger \mathfrak{P} y)}{\zeta_2(\mathfrak{P}^\dagger \mathfrak{P} y)}\mathrm{d}\widetilde{p}_1(y) = \int_{\mathbb{R}^{n(c-1)}} \log \frac{\zeta_1(y)}{\zeta_2(y)}\mathrm{d}\widetilde{p}_1(y) \tag{6.28b}$$

$$= \mathrm{KL}(\widetilde{p}_1, \widetilde{p}_2). \tag{6.28c}$$

□

Now suppose v_0 follows a normal distribution $v_0 \sim \mathcal{N}(\mu_0, \mathcal{V}_0\mathcal{V}_0^\top)$ on $T_0\mathcal{W}$. Since (6.20) is an affine transformation of v_0, we can easily compute the distribution of $v(t)$ as

$$v(t) \sim \mathcal{N}(\mu(t), \mathcal{V}(t)\mathcal{V}(t)^\top) \tag{6.29a}$$

$$\mu(t) = t\varphi(t\Pi_0^{\mathfrak{v}}\overline{A}\mathcal{R}_{w_0}^{\mathfrak{v}})\Pi_0^{\mathfrak{v}}\overline{A}w_0 + \mathrm{expm}(t\Pi_0^{\mathfrak{v}}\overline{A}\mathcal{R}_{w_0}^{\mathfrak{v}})\mu_0 \tag{6.29b}$$

$$\mathcal{V}(t) = \mathrm{expm}(t\Pi_0^{\mathfrak{v}}\overline{A}\mathcal{R}_{w_0}^{\mathfrak{v}})\mathcal{V}_0. \tag{6.29c}$$

Note that (6.29) is indeed a normal distribution on $T_0\mathcal{W}$ in the sense of Definition 6.3, because one can show that

$$\varphi(t\Pi_0^{\mathfrak{v}}\overline{A}\mathcal{R}_{w_0}^{\mathfrak{v}}) = \Pi_0^{\mathfrak{v}}\varphi(t\Pi_0^{\mathfrak{v}}\overline{A}\mathcal{R}_{w_0}^{\mathfrak{v}}) \in T_0\mathcal{W} \tag{6.30a}$$

$$\mathrm{expm}(t\Pi_0^{\mathfrak{v}}\overline{A}\mathcal{R}_{w_0}^{\mathfrak{v}}) = \Pi_0^{\mathfrak{v}}\,\mathrm{expm}(t\Pi_0^{\mathfrak{v}}\overline{A}\mathcal{R}_{w_0}^{\mathfrak{v}}) \in T_0\mathcal{W} \tag{6.30b}$$

by using the series expansions of both matrix functions and $\Pi_0^{\mathfrak{v}} \circ \Pi_0^{\mathfrak{v}} = \Pi_0^{\mathfrak{v}}$.

6.1.3 Complete Classification Architecture

The LAF construction in Section 6.1.1 does not specify a number of nodes n or graph adjacency. In image classification, a one-to-one relationship between image pixels and graph nodes is not necessarily natural because, following the reasoning of Chapter 3, the class decision being made gradually over time would be modelled on a single simplex $\mathcal{S}_c$. Here, we propose to raise the level of abstraction by choosing a moderate number n of nodes in a densely connected graph with learned interaction. The underlying reasoning is that in a difficult decision process, multiple aspects of the image in question may jointly constitute reasons to decide for any given class. An intuitive example is the presence or absence of semantic properties. Suppose we are trying to classify images into categories *cat* and *dog* and suppose we have access to n semantic properties of each image, like specific types of fur textures or facial features subject to uncertainty. The joint presence or absence of these properties is informative for the eventual class decision. It may be modelled by using the methods of Chapter 3, using n binary variables associated with nodes of a densely connected graph. In practice, we will not extract features indicative of semantic properties manually, but learn them from data as part of an end-to-end process $\widetilde{\phi}_{\mathfrak{p}} \colon \mathcal{X} \to T_0\mathcal{S}_c$. For simplicity, we will also assume that each of these abstract properties can be modelled by a discrete variable taking values in $[c]$. After fixing n as a hyperparameter, the process $\widetilde{\phi}_{\mathfrak{p}}$ consists of the following stages.

1. *Feature extraction.* A parameterized function $f_\theta \colon \mathcal{X} \to \mathcal{W}$ maps images to an initial point of (6.8), i.e. $x \mapsto f_\theta(x) = w_0$. We use a residual neural network architecture of [88] with softmax as a last layer for this purpose.

2. *Randomized LAF.* An initialization v_0 of (6.19) is drawn and the solution (6.20) after fixed time $t > 0$ is computed by approximating the required matrix function actions $\varphi(\mathsf{M})v_D$ and $\operatorname{expm}(\mathsf{M})v_0$ through the Krylov subspace method (6.17).

3. *Logit decoding.* The computed state $v(t) \in T_0\mathcal{W}$ is linearly mapped to $T_0\mathcal{S}_c$ by a learned operator $\mathsf{P} \colon T_0\mathcal{W} \to T_0\mathcal{S}_c$. In our experiments, we learn this decoding jointly with feature extraction.

The parameters $\mathfrak{p}$ of $\widetilde{\phi}_\mathfrak{p}$ are comprised of the feature extraction parameters θ, the entries of $\overline{A}$ in (6.10b), the logit decoding operator P and the tangent space initialization v_0 in (6.19). Following (6.2), we partition them according to

$$\mathfrak{p} = (\mathfrak{p}^\mathsf{d}, \mathfrak{p}^\mathsf{s}), \qquad \mathfrak{p}^\mathsf{d} = (\theta, \overline{A}, \mathsf{P}), \quad \mathfrak{p}^\mathsf{s} = v_0. \tag{6.31}$$

6.2 Risk Certification

For classification, the natural loss function is ℓ^{01}, which counts classification errors and takes values in $\{0, 1\} \subseteq [0, 1]$. The PAC-Bayes-kl inequality is a popular generalization bound in this case.

Theorem 6.5 (PAC-Bayes-kl Inequality [138]) *Let $m > 8$ denote the size of an i.i.d. sample from a data distribution and let loss take values in $[0, 1]$. For a PAC-Bayes prior π which does not have access to the sample and for all PAC-Bayes posteriors ρ it holds*

$$\operatorname{kl}(\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}_m(\mathfrak{p})], \mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}(\mathfrak{p})]) \leq \frac{1}{m}(\operatorname{KL}(\rho \colon \pi) + \log \frac{2\sqrt{m}}{\delta}) \tag{6.32}$$

with probability at least $1 - \delta$ over the draw of the sample.

Here, $\operatorname{kl}(\alpha, \beta)$ denotes the relative entropy of two Bernoulli distributions with probabilities α and β for the respective *heads* events

$$\operatorname{kl}(\alpha, \beta) = \alpha \log \frac{\alpha}{\beta} + (1 - \alpha) \log \frac{(1 - \alpha)}{(1 - \beta)}. \tag{6.33}$$

Define the pseudo inverse function in the second argument

$$\operatorname{kl}^{-1}(\alpha, \beta) = \sup \{\widetilde{\beta} \in [\alpha, 1] \colon \operatorname{kl}(\alpha, \widetilde{\beta}) \leq \beta\} \tag{6.34}$$

such that (6.32) implies

$$\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}(\mathfrak{p})] = \operatorname{kl}^{-1}\left(\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}_m(\mathfrak{p})], \frac{1}{m}(\operatorname{KL}(\rho \colon \pi) + \log \frac{2\sqrt{m}}{\delta})\right). \tag{6.35}$$

Because (6.32) bounds expected risk only implicitly, various relaxations have been developed [142, 202, 168]. We will use the one proposed by [202], presented next.

Theorem 6.6 (PAC-Bayes-λ Inequality [202]) *Under the conditions of Theorem 6.5 it holds*

$$\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}(\mathfrak{p})] \leq \frac{\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}_m(\mathfrak{p})]}{1-\frac{\lambda}{2}} + \frac{\mathrm{KL}(\rho\colon \pi) + \log\frac{2\sqrt{m}}{\delta}}{\lambda\left(1-\frac{\lambda}{2}\right)m} \tag{6.36}$$

with probability at least $1-\delta$ over the draw of the sample, for all PAC-Bayes posteriors ρ and all $\lambda \in (0,2)$ simultaneously.

Further, [67] note that (6.35) can be evaluated numerically by employing Newton iterations and [47] show that the gradient of (6.34) can be written in terms of kl^{-1} itself, making it amenable to the same numerical approximation. Irrespective of the chosen method of transforming (6.32) into a risk bound, computing expected empirical risk $\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}_m(\mathfrak{p})]$ remains a challange. Since kl^{-1} is a monotone function of its first argument, an upper bound on expected empirical risk suffices to give risk certificates via (6.32). The same is true for the relaxation (6.36). A simple way of achieving such a bound is by drawing $M > 0$ samples $\{\mathfrak{p}^{(k)}\}_{k\in[M]}$ independently from ρ and using Hoeffding's inequality which gives

$$\mathbb{E}_{\mathfrak{p}\sim\rho}\mathcal{R}_m(\mathfrak{p}) \leq \frac{1}{M}\sum_{k\in[M]}\mathcal{R}_m(\mathfrak{p}^{(k)}) + \sqrt{\frac{\log\frac{1}{\delta'}}{2M}} \tag{6.37}$$

with probability at least $1-\delta'$. An improvement over this simple method is proposed by [117], which we recite in the slightly refined version of [20].

Theorem 6.7 (Theorem 2.5 of [117]) *For samples $\{\mathfrak{p}^{(k)}\}_{k\in[M]}$ drawn independently from ρ and loss function taking values in $[0,1]$, it holds*

$$\mathrm{kl}\left(\frac{1}{M}\sum_{k\in[M]}\mathcal{R}_m(\mathfrak{p}^{(k)}),\ \mathbb{E}_{\mathfrak{p}\sim\rho}\mathcal{R}_m(\mathfrak{p})\right) \leq \frac{\log\frac{1}{\delta'}}{M} \tag{6.38}$$

with probability at least $1-\delta'$ over the draw of samples.

The bound (6.38) is indeed an improvement over (6.37), which can be seen by applying Pinsker's inequality $\mathrm{kl}(\alpha,\beta) \geq 2(\beta-\alpha)^2$ to recover (6.37) from (6.38). This method is used in multiple works, including [67] and [162]. However, evaluating the bound is computationally expensive in practice. This is because a single evaluation of $\mathcal{R}_m(\mathfrak{p}^{(k)})$ requires a forward pass for each datum used for risk certification. Recently, [21] have proposed a modification which combats this computational problem. Let $\{(x^{(i)}, y^{(i)})\}_{i\in[m]}$ denote the set of data used for risk certification, which we call *validation set*. Further, let $\{\mathfrak{p}^{(k,i)}\}_{k\in[M],i\in[m]}$ denote mM samples, independently drawn from ρ. Then [21, Theorem 5.1] shows

$$\mathbb{E}_{\mathfrak{p}\sim\rho}\mathcal{R}_m(\mathfrak{p}) = \frac{1}{m}\sum_{i\in[m]}\mathbb{E}_{\mathfrak{p}\sim\rho}\ell^{01}(\phi_{\mathfrak{p}}(x^{(i)}), y^{(i)}) \tag{6.39a}$$

$$\leq \frac{1}{mM}\sum_{k\in[M]}\sum_{i\in[m]}\ell^{01}(\phi_{\mathfrak{p}^{(k,i)}}(x^{(i)}), y^{(i)}) + \sqrt{\frac{\log\frac{1}{\delta'}}{2mM}} \tag{6.39b}$$

with probability at least $1 - \delta'$. This is based on the observation that

$$\sum_{k\in[M]} \sum_{i\in[m]} \frac{1}{mM} \ell^{01}(\phi_{\mathfrak{p}^{(k,i)}}(x^{(i)}), y^{(i)}) \tag{6.40}$$

is a sum of mM independent random variables with expectation $\mathbb{E}_{\mathfrak{p}\sim\rho}\mathcal{R}_m(\mathfrak{p})$. Since each summand is nonnegative and bounded by $\frac{1}{mM}$, Hoeffding's inequality gives (6.39). The bound (6.39) improves the rate of decay for expected empirical risk estimation by a factor of $\mathcal{O}(m)$ for the same number of forward passes. Similar to the way Theorem 6.7 improves over (6.37), (6.39) can also be sharpened.

Theorem 6.8 (Theorem 2 of [20]) *For samples $\{\mathfrak{p}^{(k,i)}\}_{k\in[M],i\in[m]}$ drawn independently from ρ and loss function taking values in $[0,1]$, it holds*

$$\mathrm{kl}\left(\frac{1}{M}\sum_{k\in[M]} \mathcal{R}_m(\mathfrak{p}^{(k,i)}),\ \mathbb{E}_{\mathfrak{p}\sim\rho}\mathcal{R}_m(\mathfrak{p})\right) \leq \frac{\log\frac{1}{\delta'}}{mM} \tag{6.41}$$

with probability at least $1 - \delta'$ over the draw of samples.

For the stochastic classifier architecture layed out in Section 6.1.3, computational efficiency can be achieved in an even more effective way, by explicitly computing the moments of classification logits, which follow a normal distribution. Recall the partition (6.31) of parameters into deterministic and stochastic subvectors and the shape of PAC-Bayesian prior and posterior distributions

$$\pi = \delta_{\mathfrak{p}^{\mathsf{d}}} \times \pi^{\mathsf{s}} \qquad \rho = \delta_{\mathfrak{p}^{\mathsf{d}}} \times \rho^{\mathsf{s}} \qquad \pi^{\mathsf{s}}, \rho^{\mathsf{s}} \in \mathcal{P}(\mathbb{R}^{d_1}) \tag{6.42}$$

from (6.3). Lemma 6.1 guarantees that these distributions are suitable within the PAC-Bayesian risk certification paradigm, provided we keep deterministic parameters fixed between prior and posterior and that π^{s} is an equivalent measure to ρ^{s}. We achieve the latter by defining both π^{s} and ρ^{s} as normal distributions

$$\pi^{\mathsf{s}} = \mathcal{N}(\mu_{\pi^{\mathsf{s}}}, \Sigma_{\pi^{\mathsf{s}}}), \qquad \Sigma_{\pi^{\mathsf{s}}} = \mathcal{V}_{\pi^{\mathsf{s}}}\mathcal{V}_{\pi^{\mathsf{s}}}^{\top} \tag{6.43a}$$

$$\rho^{\mathsf{s}} = \mathcal{N}(\mu_{\rho^{\mathsf{s}}}, \Sigma_{\rho^{\mathsf{s}}}), \qquad \Sigma_{\rho^{\mathsf{s}}} = \mathcal{V}_{\rho^{\mathsf{s}}}\mathcal{V}_{\rho^{\mathsf{s}}}^{\top} \tag{6.43b}$$

with full support on $T_0\mathcal{W}$. Using these definitions, Lemma 6.1 shows $\mathrm{KL}(\rho,\pi) = \mathrm{KL}(\rho^{\mathsf{s}}, \pi^{\mathsf{s}})$. As noted in Section 6.1.3, we decode classification logits by a linear map

$$\mathsf{P}\colon T_0\mathcal{W} \to T_0\mathcal{S}_c \tag{6.44}$$

and, given $v(t) \in \mathbb{R}^{nc}$ computed by integration of the LAF, classification logits are found as $\mathsf{P}v(t)$. Since $v(t)$ follows the normal distribution (6.29) and P is a linear operator, classification logits $\widetilde{\phi}_{\mathfrak{p}}(x)$ for each input datum $x \in \mathcal{X}$ also follow a normal distribution

$$\widetilde{\phi}_{\mathfrak{p}}(x) \sim \mathcal{N}(\mathsf{P}\mu(t), (\mathsf{P}\mathcal{V}(t))(\mathsf{P}\mathcal{V}(t))^{\top}) \tag{6.45a}$$

$$\mu(t) = t\varphi(t\Pi_0^{\mathfrak{v}}\overline{A}\mathcal{R}_{w_0}^{\mathfrak{v}})\Pi_0^{\mathfrak{v}}\overline{A}w_0 + \mathrm{expm}(t\Pi_0^{\mathfrak{v}}\overline{A}\mathcal{R}_{w_0}^{\mathfrak{v}})\mu_0 \tag{6.45b}$$

$$\mathcal{V}(t) = \mathrm{expm}(t\Pi_0^{\mathfrak{v}}\overline{A}\mathcal{R}_{w_0}^{\mathfrak{v}})\mathcal{V}_0 \tag{6.45c}$$

on $T_0\mathcal{S}_c$. To compute the moments in (6.45), we can use the Krylov subspace method described in Section 6.1.1. Let P be realized by a matrix with rows $\mathsf{p}_i \in \mathbb{R}^{nc}$, $i \in [c]$. Then

$$\mathcal{V}(t)^\top \mathsf{p}_i = \mathcal{V}_0^\top \operatorname{expm}(t\Pi_0^{\mathfrak{v}} \overline{A} \mathcal{R}_{w_0}^{\mathfrak{v}})^\top \mathsf{p}_i = \mathcal{V}_0^\top \operatorname{expm}(t\mathcal{R}_{w_0}^{\mathfrak{v}} \overline{A}^\top \Pi_0^{\mathfrak{v}}) \mathsf{p}_i. \tag{6.46}$$

Crucially, (6.46) clarifies that computing $\mathsf{P}\mathcal{V}(t)$ requires only c separate applications of Arnoldi's method. Note that

$$\operatorname{expm}\left(\begin{bmatrix} \mathsf{M} & tv_D \\ 0 & 0 \end{bmatrix}\right) \begin{bmatrix} v_0 \\ 1 \end{bmatrix} = \begin{bmatrix} \operatorname{expm}(\mathsf{M}) & t\varphi(\mathsf{M})v_D \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_0 \\ 1 \end{bmatrix} \tag{6.47}$$

by [174, Proposition 2.1], which allows to compute $\mu(t)$ by applying Arnoldi's method only once, to a slightly larger matrix. In summary, to compute the moments (6.45), feature extraction needs to be performed once and Arnoldi's method needs to be applied $c+1$ times. In the worst case, this has close to the computational complexity of $c+1$ forward passes but, depending on the complexity of feature extraction, the practical effort can be closer to a single forward pass. Define the function $\widehat{\phi}(\mathfrak{p}, \cdot) = \widetilde{\phi}_{\mathfrak{p}}$ and let $\widehat{r}\colon T_0\mathcal{S}_c \to [c]$ denote the rounding operation in (6.7). Expected empirical risk under the posterior reads

$$\begin{aligned} \mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}_m(\mathfrak{p})] &= \frac{1}{m} \sum_{i\in[m]} \int \ell(\widehat{r}(\widehat{\phi}(\mathfrak{p}, x^{(i)})), y^{(i)}) \mathrm{d}\rho(\mathfrak{p}) && \text{(6.48a)} \\ &= \frac{1}{m} \sum_{i\in[m]} \int \ell(\widehat{r}(z), y^{(i)}) \, \mathrm{d}\Big(\widehat{\phi}(\cdot, x^{(i)})_\sharp \rho\Big)(z) && \text{(6.48b)} \\ &= \frac{1}{m} \sum_{i\in[m]} \mathbb{E}_{z\sim\widehat{\phi}(\cdot,x^{(i)})_\sharp\rho}[\ell(\widehat{r}(z), y^{(i)})]. && \text{(6.48c)} \end{aligned}$$

For each $i \in [m]$, let $\{z^{(k,i)}\}_{k\in[M]}$ be $M > 0$ samples drawn independently from the normal distribution $\rho_i = \widehat{\phi}(\cdot, x^{(i)})_\sharp \rho$ specified in (6.45). Then $\ell(\widehat{r}(z^{(k,i)}), y^{(i)})$ are mM independent random variables, taking values in $[0, 1]$ and

$$\begin{aligned} \mathbb{E}\Big[\sum_{k\in[M]} \sum_{i\in[m]} \frac{1}{mM} \ell(\widehat{r}(z^{(k,i)}), y^{(i)})\Big] &= \frac{1}{m} \sum_{i\in[m]} \frac{1}{M} \sum_{k\in[M]} \mathbb{E}_{z^{(k,i)}\sim\rho_i} \ell(\widehat{r}(z^{(k,i)}), y^{(i)}) && \text{(6.49a)} \\ &= \frac{1}{m} \sum_{i\in[m]} \mathbb{E}_{z\sim\rho_i} \ell(\widehat{r}(z), y^{(i)}) && \text{(6.49b)} \\ &= \mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}_m(\mathfrak{p})] && \text{(6.49c)} \end{aligned}$$

by (6.48). Thus, Hoeffding's inequality gives

$$\mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}_m(\mathfrak{p})] \leq \frac{1}{mM} \sum_{k\in[M]} \sum_{i\in[m]} \ell(\widehat{r}(z^{(k,i)}), y^{(i)}) + \sqrt{\frac{\log \frac{1}{\delta'}}{mM}} \tag{6.50}$$

with probability at least $1 - \delta'$ over the draw of samples. In contrast to the approach of Theorem 6.8, drawing M samples $\{z^{(k,i)}\}_{k\in[M]}$ does not require any forward passes through the classifier. Instead, the main computational effort required to evaluate (6.50) lies in

computing the moments of $\widehat{\phi}(\cdot, x^{(i)})_\sharp \rho$ specified in (6.45) for each datum. As outlined above, this is bounded by the cost of $\mathcal{O}(mc)$ forward passes in the worst case. Thus, by using the method layed out in Section 6.1.3, we achieve the same bound on expected empirical risk as (6.39) at the cost of $\mathcal{O}(mc)$ forward passes, instead of $\mathcal{O}(mM)$.

Due to the comparatively low dimension of the integrand in (6.48), Quasi-Monte-Carlo (QMC) [57, 58] methods are a promising alternative, with empirical support presented in [25]. The basic idea is to replace randomly sampled integration points by point sets with low *discrepancy*, which often leads to improved convergence rates. A central result on QMC methods is the Koksma-Hlawka inequality [58, Theorem 3.9] which states that the integration error of a QMC method is bounded by the discrepancy of the integration points, multiplied by the variation of the integrand in the sense of Hardy and Krause [85, 113]. Unfortunately, the Hardy-Krause variation is unbounded for indicator functions of many sets other than axis-aligned hyperrectangles [154, Proposition 24]. This is an obstacle for the application of QMC methods in the case at hand because, for 0/1 loss function, the integrand (6.48) is an indicator function of a typically more general set. Thus, further study is needed to apply QMC methods in the present setting. First, different notions of variation for multivariate functions, such as the one proposed in [2], with corresponding generalized Koksma-Hlawka inequality are required. In addition, speciallized methods need to be developed in order to match the fast rate $\mathcal{O}(\sqrt{mM})$ of Monte-Carlo estimators like (6.50), which leverage the independence structure of individual terms in the empirical risk.

6.3 Experiments

A common pattern in the practical implementation of PAC-Bayesian risk certification methods is to split the available data into a *training* and a *validation* set [162, 47, 160]. The training set is used to learn a data-dependent PAC-Bayesian prior π, by first training a deterministic classifier through empirical risk minimization and subsequently randomizing the learned parameters. The posterior ρ can then be initialized at the prior and further trained on all available data (including validation data) by minimizing a high-probability bound on its *risk*.

As empirical support for the applicability of our self-certified classification approach, we perform image classification on CIFAR-10 [115] and FashionMNIST [216], following the above methodology. We choose a graph of $n = 50$ nodes with dense and symmetric adjacency matrix Ω. PAC-Bayes prior π and posterior ρ are chosen as distributions with structure (6.42), implemented in the manner described in Section 6.1.3. Partition of parameters in deterministic and stochastic subvectors is performed according to (6.43). Stochastic parameters are randomized LAF initializations following a normal distribution (6.43). We fix both π^{s} and ρ^{s} to have zero mean, i.e. $\mu_{\pi^{\mathsf{s}}} = \mu_{\rho^{\mathsf{s}}} = 0 \in T_0\mathcal{W}$. Prior covariance $\Sigma_{\pi^{\mathsf{s}}}$ is fixed and defined by

$$\Sigma_{\pi^{\mathsf{s}}} = \mathcal{V}_{\pi^{\mathsf{s}}} \mathcal{V}_{\pi^{\mathsf{s}}}^\top, \qquad \mathcal{V}_{\pi^{\mathsf{s}}} = 2\mathfrak{P}\mathbb{I}_{n(c-1)} \tag{6.51}$$

with an orthogonal basis $\mathfrak{P}$ for the tangent space $T_0\mathcal{W}$ as defined in (6.23).

A validation set of $m = 10^4$ data is split off from the predefined training dataset of CIFAR-10 and FashionMNIST respectively. The remaining training data (40k for CIFAR-10 and 50k for FashionMNIST) are used to train deterministic parameters by minimizing empirical risk. To this end, we use stochastic gradient descent with batch size 128, (initial) learning rate 0.01, momentum 0.9 and weight decay 10^{-3} over 200 epochs with a cosine annealing schedule [129] and light data augmentation regime as in [224].

Initializing ρ at π, we subsequently train ρ^{s} by minimizing the r.h.s. of the bound (6.36). To this end, we initially fix λ and use cross-entropy as a differentiable surrogate loss. In order to compute gradients for the moments of ρ^{s}, we employ the reparameterization trick [107] of representing ρ^{s} as pushforward of a standard normal on $\mathbb{R}^{n(c-1)}$. For optimization, we use stochastic gradient descent with learning rate 0.5 and momentum 0.9 over 80 epochs. To improve training signal, stochastic gradients of expected empirical risk relative to the surrogate loss are accumulated over a whole epoch.

For final risk certification, we bound expected empirical risk under the posterior relative to 0/1 loss, by explicitly computing the pushforward $\rho_i = \widehat{\phi}(\cdot, x^{(i)})_\sharp \rho$ specified in (6.45) for each validation datum $x^{(i)}$ and employing (6.50) with $M = 15$. Since increasing M does not result in significant computational cost when using the described approach, the bound can easily be improved, but we choose $M = 15$ to match the statistical guarantee of [162]. The result of this benchmark is summarized in Table 6.1, revealing that our method is able to achieve strong stochastic classifiers with tightly bounded classification risk.

In order to compare more directly to [162], we additionally train a classifier on CIFAR-10 with features extracted by the 9-layer CNN of [162] and corresponding simple SGD training regime (70 epochs, learning rate 0.01, momentum 0.95, dropout rate 0.2) without data augmentation. The result of this comparison is summarized in Table 6.2, where we use the term *PAC-Bayes by backprop* (PBB) to refer to the approach of [162]. Unfortunately, we were unable to use the larger CNN feature extractors proposed in [162] due to vanishing gradient issues when training deterministic classifiers. However, training is stable for ResNet18 features which, by comparison of deterministic CIFAR-10 classification test error between Tables 6.1 and 6.2, appear much stronger.

Our deterministic classifier performs slightly worse than PBB, likely due to a lack of hyperparameter tuning. In turn, after optimizing the bound (6.36), our posterior slightly improves on PBB in terms of expected empirical risk and we achieve a slightly lower bound on generalization risk. Tightness of the PBB certificate, as measured by the difference between expected empirical risk of the posterior compared to the risk bound, is slightly better than ours. We conclude that our method is computationally efficient and able to achieve empirical performance and risk bounds which are on par with PBB. [162] employ Theorem 6.7 to bound expected empirical risk, which requires many GPU hours to compute $\mathcal{O}(m^2 M)$ forward passes. For the datasets at hand, this can be improved by multiple orders of magnitude through the approach of Theorem 6.8, requiring $\mathcal{O}(mM)$ forward passes for the same statistical guarantee. As outlined above, our model produces a normal distribution of classification logits for each datum, whose moments can be computed at computational cost bounded by c forward passes. Thus, bounding expected empirical risk by (6.50) further improves on direct application of Theorem 6.8, with computational cost bounded by $\mathcal{O}(mc)$ forward passes. For the data at hand, the final compute time required

	Deterministic	Prior	Posterior	Certificate
CIFAR-10	5.78	8.10	5.82	7.53
FashionMNIST	4.74	6.03	4.82	6.44

Table 6.1: Benchmark results of LAF classifiers with ResNet18 features on CIFAR-10 and FashionMNIST. Out-of-sample error (%) of the deterministic LAF classifier, as well as expected empirical risks for prior and posterior are evaluated on 10k held-out test data. The risk certificate is the bound (6.36) on expected posterior risk for optimized λ and error probability bounded by $\delta + \delta' = 0.035$.

	Deterministic	Prior	Posterior	Certificate	Tightness
LAF	20.32	22.09	**20.63**	**23.24**	2.61
PBB [162]	**19.46**	**21.69**	21.61	23.77	**2.16**

Table 6.2: Benchmark results of LAF classifiers employing the 9-layer CNN features of [162]. Out-of-sample error (%) of the deterministic LAF classifier, as well as expected empirical risks for prior and posterior are evaluated on 10k held-out test data. The risk certificate is the bound (6.36) on expected posterior risk for optimized λ and error probability bounded by $\delta + \delta' = 0.035$.

to bound expected empirical risk is under 10 seconds on a single GPU.

For posterior training, we use cross-entropy as a differentiable surrogate loss. This appears problematic because the bound (6.36) only certifies risk w.r.t. bounded loss functions. [162] address this by modifying cross-entropy to obtain a closely related bounded loss function which is amenable to risk certification. We do not perform this modification and therefore do not obtain valid risk certificates for surrogate loss. However, for classification the bound (6.36) holds for *all* posterior distributions, regardless of how they have been computed. Therefore, using unbounded surrogate loss for training does not touch the validity of risk certificates for the bounded 0/1 loss reported in Tables 6.2 and 6.1. Accordingly, no certificate for surrogate loss is reported.

7 Certified Structured Prediction

In Chapters 4 and 5, we have reasoned about the difficulty of structured prediction geometrically, by pointing to the combinatorial dimension of $\mathcal{S}_N$ and constructing lower-dimensional models to approximate complex joint distributions. Universal feasibility of such approximations can not be assumed, but the empirical examples presented in Chapter 5 illustrate viability of the approach for many data distributions of interest in practical applications.

We now turn to statistical aspects of structured prediction and pose the task of learning a joint data distribution as composed of learning a structured predictor and constructing high-probability generalization bounds, like the PAC-Bayesian bounds for classification in Chapter 6. However, because joint data distributions have rich internal structure which does not factorize into independent components, the independence assumption underlying most established PAC-Bayesian bounds does not hold. In some settings, even though data may still be available in the form of multiple independent draws from a joint distribution, generalization bounds converge slowly compared to the effort of label aquisition. For instance, one may expect that pixel-wise segmentation of an image contains rich information to be exploited in supervised learning. However, a generalization bound like Theorem 6.5, which converges merely in the number of *independent* segmented images, is unable to leverage this effectively. Consider the extreme case of a single, very large segmented image, depicting content comparable to an entire semantic segmentation dataset. Irrespective of its size, Theorem 6.5 never predicts generalization of a PAC-Bayesian posterior trained on this image, because it only constitutes a single draw of data from a complex joint distribution of many (pixel and label) variables. This construction may appear artificial, but similar statistical dependencies appear in applications such as graph node classification, motivating the development of statistical learning theories which account for generalization from a single example. Addressing this point in particular, [126] presents an analysis of dependency structure between variables and proves a risk certificate which decays in both the number of structured examples *m and their size d*. Refering back to the example of image segmentation, this amounts to a high-probability bound on the fraction of misslabeled pixels which *decays with the number of labeled pixels* observed during training as opposed to merely the number of segmented images.

Building on this work, we present a novel PAC-Bayesian risk bound for structured prediction wherein the rate of generalization scales not only with the number of structured

examples but also with their size. The underlying assumption, conforming to ongoing research on generative models, is that data are generated by the Knothe-Rosenblatt rearrangement of a factorizing reference measure. This allows to explicitly distill the structure between random output variables into a Wasserstein dependency matrix. Our work makes a preliminary step towards leveraging powerful generative models to establish generalization bounds for discriminative downstream tasks in the challenging setting of structured prediction.

Recall from Section 2.2 that the concentration of measure phenomenon is at the core of statistical learning theory. It posits that a stable function of a large number of weakly dependent random variables will take values close to its mean [119, 31]. This is relevant because model risk, the expected loss on unseen data, is the mean of empirical risk under the draw of the sample which allows to build learning theories on concentration of measure results.

The popularity of PAC-Bayesian methods is partly due to their minimal assumptions on the data distribution in classification settings. Nevertheless, we share the sentiment of previous authors [126] that some assumption on the data-generating process is required in structured prediction. This is because the distribution of data conditioned on a fixed set of values for a subset of variables is central to establishing concentration of measure via the martingale method [111]. Consider again the example of image segmentation. Once we have fixed a sufficiently large number of pixels to arbitrary values (and class labels), even a large dataset will not contain an abundance of data which match these values and thus provide statistical power to learn the conditional distribution. This problem is well-known in conditional density estimation [206]. As a remedy, we propose to use a triangular and monotone transport, a *Knothe-Rosenblatt (KR) rearrangement* [108, 171, 37, 29, 136] of a reference measure as data model. This choice is attractive for multiple reasons. First, any data distribution which does not contain atoms can be represented uniquely in this way [23] which should suffice to represent many distributions of practical interest. In particular, any data distribution which is absolutely continuous with respect to the Lebesgue measure satisfies this requirement. With regard to conditional distributions, the KR-rearrangement has the convenient property that conditioning on a fixed value for a subset of variables can again be represented by KR-rearrangement. We will use this property in our construction of coupling measures between conditional distributions. We stress the fact that many established approaches to generative modelling can be seen as instances of measure transport. For instance, it includes normalizing flows [198, 197, 109, 157, 173], diffusion models [194, 90], generative adversarial networks and variational autoencoders [32, 74]. While most measure transport models which currently enjoy empirical success are not KR-rearrangements, we hope that the methods presented here can lay the foundation of leveraging powerful generative models to build risk certificates for discriminative downstream tasks.

To this end, we distill relevant structure of the data distribution into a Wasserstein dependency matrix. Our analysis hinges on state-of-the-art results in concentration theory [111] which serve to bound moment-generating functions by properties of the Wasserstein dependency matrix. Finally, we invoke a PAC-Bayesian argument to derive the desired risk certificate.

Additional Notation In addition to the basic notation introduced in Section 1.3, we require the following, more speciallized notions. If $x \in \mathcal{Z}^d$ is a vector, we refer to the subvector of entries with index in a set $I \subseteq [d]$ as x^I. In particular, index sets of interest will be half-open and closed intervals $(i, d] \subseteq [d]$ and $[i, d] \subseteq [d]$. Analogously, we will index the output of vector-valued functions f^I and marginal measures μ^I. For a set $\mathcal{B} \subseteq \mathcal{Z}^d$, we denote its complement in $\mathcal{Z}^d$ by $\mathcal{B}^c = \mathcal{Z}^d \setminus \mathcal{B}$ and for a measure μ on $\mathcal{Z}^d$, we denote the conditional measure given $\mathcal{B}^c$ as $\mu|\mathcal{B}^c$. If X is a random variable with distribution μ on $\mathcal{Z}^d$ and $I, J \subseteq [d]$ are disjoint index sets with $I \cup J = [d]$, we denote the conditional law of X^I given $X^J = x^J$ as $\mu(dx^I|x^J)$.

Inkeeping with the notation of Section 2.2, we use the symbols $\mathcal{X}$ and $\mathcal{Y}$ to respectively denote an input space and output space. For the structured prediction setting, we assume that μ is a distribution on $\mathcal{Z}^d = (\mathcal{X} \times \mathcal{Y})^d$. There are two restrictions inherent to this setup. First, an input is always paired with an output and thus the number of inputs needs to match the number of outputs. Second, all structured data will be drawn from μ and thus the size of each structured datum will be the same. Otherwise, $\mathcal{X}$ and $\mathcal{Y}$ can in principle be arbitrary sets which admit metrics. For concreteness, think of $\mathcal{X} = [0, 1] \subseteq \mathbb{R}$ as being a set of gray values and $\mathcal{Y} = \mathbb{R}$ containing signed distances from a semantic boundary [153] in an image with d pixels. In this case, $\mathcal{Z}^d$ contains all binary segmentations of grayvalue images.

Given a sample $\mathcal{D}_m = (X^{(k)}, Y^{(k)})_{k\in[m]}$ drawn from μ^m, we will again follow the PAC-Bayesian paradigm and consider stochastic predictors ρ, i.e. measures on a hypothesis space $\mathcal{H}$ of predictors $\phi_{\mathfrak{p}} \colon \mathcal{X}^d \to \mathcal{Y}^d$ as identified with measures on the underlying parameter space Θ from which $\mathfrak{p}$ is selected. The goal is to bound *expected risk* of the posterior ρ

$$\mathcal{R}(\rho) = \mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}(\mathfrak{p})], \tag{7.1}$$

by tractable quantities, such as the *expected empirical risk*

$$\mathcal{R}_m(\rho, \mathcal{D}_m) = \mathbb{E}_{\mathfrak{p}\sim\rho}[\mathcal{R}_m(\mathfrak{p}, \mathcal{D}_m)]. \tag{7.2}$$

We further assume that the loss of structured outputs is the mean of bounded *pointwise* loss $\ell \colon \mathcal{Y} \times \mathcal{Y} \to [0, 1]$

$$L(\gamma^{(k)}, y^{(k)}) = \frac{1}{d} \sum_{i\in[d]} \ell(\gamma_i^{(k)}, y_i^{(k)}). \tag{7.3}$$

We will invoke a line of reasoning put forward in [111] and propose a novel approach to structured prediction based on the measure-transport framework outlined in Section 1. To this end, we first define the following formal notions of *stability* and *dependence*. Let σ be a metric such that $\mathcal{Z}$ has finite diameter

$$\|\sigma\| = \sup_{z,z'\in\mathcal{Z}} \sigma(z, z') < \infty \tag{7.4}$$

and let $\sigma^d(z, z') = \sum_{i\in[d]} \sigma(z_i, z_i')$ denote the corresponding product metric on $\mathcal{Z}^d$.

Definition 7.1 (Local oscillation) *Let $f\colon \mathcal{Z}^d \to \mathbb{R}$ be Lipschitz with respect to σ^d. Then the quantities*

$$\chi_i(f) = \sup_{z,z' \in \mathcal{Z}^d, z'_{[d]\setminus\{i\}} = z_{[d]\setminus\{i\}}} \frac{|f(z) - f(z')|}{\sigma(z_i, z'_i)}, \qquad i \in [d] \tag{7.5}$$

are called the local oscillations of f.

The vector of local oscillations gives a granular account of stability. In order to discuss interdependence of data in a probability space $(\mathcal{Z}^d, \mu, \Sigma)$, define the Markov kernels

$$K^{(i)}(x, dy) = \delta_{x^{[i-1]}}(dy^{[i-1]}) \otimes \mu^{[i,d]}(dy^{[i,d]}|x^{[i-1]}), \qquad i \in [d] \tag{7.6}$$

as well as $K^{(d+1)}(x, dy) = \delta_x(dy)$ and their action on functions

$$K^{(i)}f(x) = \int f(y)K^{(i)}(x, dy) = \int f(x^{[i-1]}y^{[i,d]})\mu^{[i,d]}(dy^{[i,d]}|x^{[i-1]}) \tag{7.7}$$

where in the edge case $i = 1$, the condition on $x^{[i-1]} = x^{\{\}}$ is removed. It turns out that the effect of the kernel (7.6) on local oscillations serves to quantify dependence of data with joint distribution μ.

Definition 7.2 (Wasserstein matrix) *For $i \in [d+1]$, let $K^{(i)}$ denote the Markov kernel (7.7). A matrix $V^{(i)} \in \mathbb{R}^{d\times d}_{\geq 0}$ is called a Wasserstein matrix [69] for $K^{(i)}$, if*

$$\chi_k(K^{(i)}f) \leq \sum_{j\in[d]} V^{(i)}_{kj}\chi_j(f), \qquad \forall\, k \in [d] \tag{7.8}$$

for any function $f\colon \mathcal{Z}^d \to \mathbb{R}$ which is Lipschitz with respect to σ^d.

The two concepts defined above will be used in Section 7.2 to construct a moment-generating function bound via the martingale method.

7.1 Triangular Measure Transport

Suppose a structured output is composed of $d > 0$ unstructured data in a space $\mathcal{Z}$. Then the target measure μ of interest is a measure on $\mathcal{Z}^d$ which does not factorize into simpler distributions. A popular method of representing complex joint distributions of interdependent random variables is to define a map $\Upsilon\colon \mathcal{Z}^d \to \mathcal{Z}^d$ which transports a tractable *factorizing* reference measure ν^d to the target measure μ, i.e. $\Upsilon_\sharp\nu^d = \mu$. This abstract framework encompasses many generative models such as normalizing flows [198, 197, 109, 157, 173], diffusion models [194, 90], generative adversarial networks and variational autoencoders [32, 74]. Here, we focus on transport maps Υ which are monotone and triangular in the sense that $\Upsilon(z)_i$ only depends on the inputs $z^{[i]}$ and each $\Upsilon(z^{[i-1]}, \cdot)_i$ is an increasing function. Such a map is called a *Knothe-Rosenblatt (KR) rearrangement* [108, 171, 37, 29, 136]. If both ν^d and μ have no atoms then the KR rearrangement exists and is unique [23]. In particular, normal distribution ν^d and any absolutely continuous (with respect to the Lebesgue measure) distribution μ meet these criteria. The KR rearrangement has the useful property that certain conditional distributions have a simple representation.

Lemma 7.3 (Lemma 1 of [136]) *Let $\Upsilon\colon \mathcal{Z}^d \to \mathcal{Z}^d$ be the KR-rearrangement which satisfies $\Upsilon_\sharp \nu^d = \mu$. For arbitrary $i \in [d]$, let $z^{[i]} \in \mathcal{Z}^k$ be fixed. Then*

$$\mu(dy^{(i,d]}|z^{[i]}) = \Upsilon^{(i,d]}(\overline{z}^{[i]}, \cdot)_\sharp \nu^{d-i} \tag{7.9}$$

where $\overline{z}^{[i]}$ is the unique element of $\mathcal{Z}^i$ such that $\Upsilon^{[i]}(\overline{z}^{[i]}) = z^{[i]}$.

Recently, numerical realization of KR rearrangements has received attention [13] and more broadly, a variety of triangular transport architectures exist [59, 60]. However, we do not focus on numerical considerations in the present theoretical work.

One may wonder if data generated by KR-rearrangement implicitly restricts the choice of possible input and output spaces $\mathcal{X}$, $\mathcal{Y}$ since the monotonicity requirement on Υ can only be satisfied if the underlying sets are ordered. For what follows, many more general input and output spaces are still permissible. Suppose, for instance, $\mathcal{X} = [0,1]^3$ contains RGB color values. By identifying $\mathcal{X}^d = [0,1]^{3d}$, we can still construct KR-rearrangements and all subsequent results hold analogously.

Lemma 7.4 (Conditioned Transport) *Let $\Upsilon\colon \Omega \to \Omega$ be a measurable function on a measurable space (Ω, Σ) and let ν, μ be measures on Ω with $\Upsilon_\sharp \nu = \mu$. Let $B \in \Sigma$ be a fixed set with $\mu(B) > 0$ and $A = \Upsilon^{-1}(B)$ its preimage under Υ. Then*

$$\Upsilon_\sharp(\nu|A) = \mu|B. \tag{7.10}$$

Proof. Let $S \in \Sigma$ be arbitrary and let $\widetilde{\mu} = \Upsilon_\sharp(\nu|A)$. Then

$$\widetilde{\mu}(S) = (\nu|A)(\Upsilon^{-1}(S)) = \frac{\nu(\Upsilon^{-1}(S) \cap A)}{\nu(A)} \tag{7.11}$$

as well as

$$(\mu|B)(S) = \frac{\mu(S \cap B)}{\mu(B)} = \frac{\nu(\Upsilon^{-1}(S \cap B))}{\nu(A)}. \tag{7.12}$$

Note that

$$\begin{aligned} x \in \Upsilon^{-1}(S) \cap \Upsilon^{-1}(B) &\Leftrightarrow \Upsilon(x) \in S \wedge \Upsilon(x) \in B && \text{(7.13a)} \\ &\Leftrightarrow \Upsilon(x) \in S \cap B && \text{(7.13b)} \\ &\Leftrightarrow x \in \Upsilon^{-1}(S \cap B) && \text{(7.13c)} \end{aligned}$$

which implies $\Upsilon^{-1}(S) \cap \Upsilon^{-1}(B) = \Upsilon^{-1}(S \cap B)$ and consequently $\widetilde{\mu}(S) = (\mu|B)(S)$. Since S was arbitrary, this shows the assertion. □

The following theorem, a generalization of Theorem 2.11, exists in various forms in the literature. To make this thesis self-contained, we recite the version in [111] which is used to bound moment-generating functions in Proposition 7.6. Note that we only use the MGF bound (7.14) in our analysis. However, the concentration inequality (7.15) also holds analogously under the assumptions of Proposition 7.6 which may be of independent interest.

Theorem 7.5 (Azuma-Hoeffding [111, Theorem 4.1]) *Let $(M^{(i)})_{i\in[m]}$ be a martingale difference sequence with respect to a filtration $(\Sigma_i)_{i\in[m]}$ of sigma algebras. Suppose that for each $i \in [m]$ there exist Σ_{i-1}-measurable random variables $A^{(i)}$, $B^{(i)}$ such that $A^{(i)} \leq M^{(i)} \leq B^{(i)}$ almost surely. Then for all $\lambda \in \mathbb{R}$ it holds that*

$$\mathbb{E}\Big[\exp\Big(\lambda \sum_{i\in[m]} M^{(i)}\Big)\Big] \leq \exp\Big(\frac{\lambda^2}{8}\sum_{i\in[m]} \|B^{(i)} - A^{(i)}\|_\infty^2\Big) \tag{7.14}$$

and consequently, for any $t \geq 0$

$$\mathbb{P}\Big(\Big|\sum_{i\in[m]} M^{(i)}\Big| \geq t\Big) \leq 2\exp\Big(-\frac{2t^2}{\sum_{i\in[m]}\|B^{(i)} - A^{(i)}\|_\infty^2}\Big). \tag{7.15}$$

7.2 PAC-Bayesian Risk Certificate

In this section we present a novel PAC-Bayesian risk bound for structured prediction which combines three main ingredients.

(1) A concentration of measure theorem for dependent data (Theorem 7.6) which builds on the notion of a Wasserstein dependency matrix;

(2) a simple construction of coupling measures between conditional distributions (Lemma 7.7) which serves to represent the Wasserstein dependency matrix;

(3) a PAC-Bayesian argument (Theorem 7.9) employing Donsker-Varadhan's variational formula in concert with concentration of measure results.

The first theorem summarizes key results from [111] on the concentration of measure phenomenon for dependent random variables. We have slightly generalized by augmenting the underlying Doob martingale construction with the inclusion of a set $\mathcal{B}$ of *bad inputs*. For inputs in this set, data stability requirements do not necessarily hold. We call the complement $\mathcal{B}^c = \mathcal{Z}^d \setminus \mathcal{B}$ the set of *good inputs*. The concept of good and bad inputs as well as related proof techniques were originally proposed by [126]. Here, we incorporate them into the more general concentration of measure formalism of [111].

Theorem 7.6 (Moment-generating function (MGF) bound for good inputs) *Let $\mathcal{B} \subseteq \mathcal{Z}^d$ be a measurable set of bad inputs. Suppose for each $i \in [d+1]$, $V^{(i)}$ is a Wasserstein matrix for the Markov kernel $K^{(i)}$ defined in* (7.6) *on the set of good inputs, that is*

$$\chi_k(K^{(i)}\widetilde{f}) \leq \sum_{j\in[d]} V_{kj}^{(i)}\chi_j(\widetilde{f}), \qquad \forall\, k \in [d] \tag{7.16}$$

for all Lipschitz (with respect to σ^d) functions $\widetilde{f}\colon \mathcal{B}^c \to \mathbb{R}$. Define the Wasserstein dependency *matrix*

$$\Gamma \in \mathbb{R}^{d\times d}, \qquad \Gamma_{ij} = \|\sigma\| V_{ij}^{(i+1)} \tag{7.17}$$

Then for all Lipschitz functions $f\colon \mathcal{Z}^d \to \mathbb{R}$, the following MGF bound holds

$$\mathbb{E}_{z\sim\mu|\mathcal{B}^c}\Big[\exp\Big(\lambda(f(z) - \mathbb{E}_{\mu|\mathcal{B}^c} f)\Big)\Big] \leq \exp\Big(\frac{\lambda^2}{8}\|\Gamma\chi(f)\|_2^2\Big). \tag{7.18}$$

Proof. For any $i \in [d]$ and $x \in \mathcal{Z}^d$ define

$$M^{(i)} = \mathbb{E}_{X\sim\mu}[f(X)|\mathcal{B}^c, X^{[i]} = x^{[i]}] - \mathbb{E}_{X\sim\mu}[f(X)|\mathcal{B}^c, X^{[i-1]} = x^{[i-1]}] \tag{7.19}$$

with the edge case

$$M^{(1)} = \mathbb{E}_{X\sim\mu}[f(X)|\mathcal{B}^c, X_1 = x_1] - \mathbb{E}_{X\sim\mu}[f(X)|\mathcal{B}^c]. \tag{7.20}$$

Due to $\mathbb{E}_{X\sim\mu}[f(X)|\mathcal{B}^c, X = x] = f(x)$ for $x \in \mathcal{B}^c$ we have

$$f - \mathbb{E}_{X\sim\mu}[f(X)|\mathcal{B}^c] = \sum_{i=1}^{d} M^{(i)}. \tag{7.21}$$

Since the conditions $X^{[i]} = x^{[i]}$ generate a nested sequence of σ-algebras, the quantities $K^{(i+1)}f(x) = \mathbb{E}_\mu[f(X)|\mathcal{B}^c, X^{[i]} = x^{[i]}]$ are a Doob martingale and (7.19) is a martingale difference sequence. In order to bound the moment generating function of f, we will bound every $M^{(i)}$ from above and below and apply the Azuma-Hoeffding theorem 7.5. We have

$$\begin{aligned}
M^{(i)} &= \mathbb{E}_\mu[f(X)|\mathcal{B}^c, X^{[i]} = x^{[i]}] - \mathbb{E}_\mu[f(X)|\mathcal{B}^c, X^{[i-1]} = x^{[i-1]}] && \text{(7.22a)}\\
&= \mathbb{E}_\mu[f(X)|\mathcal{B}^c, X^{[i]} = x^{[i]}] \\
&\quad - \mathbb{E}_\mu[\mathbb{E}_\mu[f(X)|\mathcal{B}^c, X^{[i-1]} = x^{[i-1]}, X_i]|\mathcal{B}^c, X^{[i-1]} = x^{[i-1]}] && \text{(7.22b)}\\
&= \int f(x^{[i]}y^{(i,d]})\mu(dy^{(i,d]}|x^{[i]}, \mathcal{B}^c) \\
&\quad - \int \Big(\int f(x^{[i]}u^{(i,d]})\mu(du^{(i,d]}|x^{[i-1]}, y_i, \mathcal{B}^c)\Big)\mu(dy^{[i,d]}|x^{[i-1]}, \mathcal{B}^c) && \text{(7.22c)}
\end{aligned}$$

by the tower property of conditional expectations. Because $\mu(dy^{[i,d]}|x^{[i-1]}, \mathcal{B}^c)$ is a probability measure, it holds

$$\int f(x^{[i]}y^{(i,d]})\mu(dy^{(i,d]}|x^{[i]}, \mathcal{B}^c) = \int \Big(\int f(x^{[i-1]}x_i u^{(i,d]})\mu(du^{(i,d]}|x^{[i]}, \mathcal{B}^c)\Big)\mu(dy^{[i,d]}|x^{[i-1]}, \mathcal{B}^c) \tag{7.23}$$

and we find

$$\begin{aligned}
M^{(i)} = \int \mu(dy^{[i,d]}|x^{[i-1]}, \mathcal{B}^c)\Big(&\int f(x^{[i-1]}x_i u^{(i,d]})\mu(du^{(i,d]}|x^{[i]}, \mathcal{B}^c) \\
&- \int f(x^{[i]}u^{(i,d]})\mu(du^{(i,d]}|x^{[i-1]}, y_i, \mathcal{B}^c)\Big). \qquad \text{(7.24)}
\end{aligned}$$

Now bound $A^{(i)} \le M^{(i)} \le B^{(i)}$ almost surely with

$$\begin{aligned}
A^{(i)} = \int \mu(dy^{[i,d]}|x^{[i-1]}, \mathcal{B}^c) \inf_{x_i \in \mathcal{B}_i^c(x^{[i-1]})} \Big(&\int f(x^{[i-1]}x_i u^{(i,d]})\mu(du^{(i,d]}|x^{[i]}, \mathcal{B}^c) \\
&- \int f(x^{[i]}u^{(i,d]})\mu(du^{(i,d]}|x^{[i-1]}, y_i, \mathcal{B}^c)\Big) \qquad \text{(7.25a)}\\
B^{(i)} = \int \mu(dy^{[i,d]}|x^{[i-1]}, \mathcal{B}^c) \sup_{x_i \in \mathcal{B}_i^c(x^{[i-1]})} \Big(&\int f(x^{[i-1]}x_i u^{(i,d]})\mu(du^{(i,d]}|x^{[i]}, \mathcal{B}^c) \\
&- \int f(x^{[i]}u^{(i,d]})\mu(du^{(i,d]}|x^{[i-1]}, y_i, \mathcal{B}^c)\Big) \qquad \text{(7.25b)}
\end{aligned}$$

where $\mathcal{B}_i^c(x^{[i-1]})$ contains all $x_i \in \mathcal{Z}$ such that there exist $x^{(i,d]} \in \mathcal{Z}^{d-i}$ with $(x^{[i-1]}, x_i, x^{(i,d]}) \in \mathcal{B}^c$. Because every realization of a random variable conditioned on $\mathcal{B}^c$ is in the set of good inputs, the difference $\|B^{(i)} - A^{(i)}\|_\infty$ can be written as

$$\sup_{x,z \in \mathcal{B}^c, x^{[d]\setminus\{i\}} = z^{[d]\setminus\{i\}}} \int f(x^{[i]}u^{(i,d]})\mu(du^{(i,d]}|x^{[i]}, \mathcal{B}^c) - \int f(z^{[i]}u^{(i,d]})\mu(du^{(i,d]}|z^{[i]}, \mathcal{B}^c) \quad (7.26)$$

By seeing this expression in terms of oscillation of the kernel action $K^{(i+1)}f$, we find

$$\|B^{(i)} - A^{(i)}\|_\infty \leq \|\sigma\|\chi_i(K^{(i+1)}\widetilde{f}) \leq \|\sigma\|(V^{(i+1)}\chi(\widetilde{f}))_i = (\Gamma\chi(\widetilde{f}))_i \quad (7.27)$$

where $\widetilde{f}\colon \mathcal{B}^c \to \mathbb{R}$ is the restriction of f to $\mathcal{B}^c$. The assertion then follows from the Azuma-Hoeffding theorem 7.5. □

An upper bound on the moment generating function will be used in the PAC-Bayesian argument concluding this section. The function f in question will be the loss of a structured datum z. Regarding (7.18), our goal is to bound the norm $\|\Gamma\chi(f)\|_2^2$ through properties of the data distribution. We will use the fact that data is represented by measure transport to establish such a bound after the following preparatory lemma.

Lemma 7.7 (Coupling from transport) *Let ν^d be a reference measure on $\mathcal{Z}^d$ and $F, G\colon \mathcal{Z}^d \to \mathcal{Z}^d$ be measurable maps. Define the map (F, G) by*

$$(F, G)\colon \mathcal{Z}^d \to \mathcal{Z}^d \times \mathcal{Z}^d, \qquad z \mapsto (F(z), G(z)) \quad (7.28)$$

Then $(F, G)_\sharp\nu^d$ is a coupling of $F_\sharp\nu^d$ and $G_\sharp\nu^d$.

Proof. Let $A \subseteq \mathcal{Z}^d$ be measurable, then

$$(F, G)_\sharp\nu^d(A, \mathcal{Z}^d) = \nu^d((F, G)^{-1}(A, \mathcal{Z}^d)) = \nu^d(F^{-1}(A)) = F_\sharp\nu^d(A) \quad (7.29)$$

which shows that $F_\sharp\nu^d$ is the first marginal of $(F, G)_\sharp\nu^d$. An analogous argument for the second marginal shows the assertion. □

By assuming μ to be represented via KR-rearrangement of a factorizing reference measure, Lemma 7.3 gives an explicit representation of KR-rearrangement for conditional distributions. From there, we invoke Lemma 7.7 to construct a coupling between conditional distributions and subsequently follow a line of reasoning put forward in [111] to explicitly construct Wasserstein matrices for the kernels (7.6) which yield a bound on (7.18) by Theorem 7.6. This leads to the following proposition.

Proposition 7.8 (Wasserstein dependency matrix from KR-rearrangement) *Let $(\mathcal{Z}^d, \Sigma, \mu)$ be a probability space with $\mu = \Upsilon_\sharp\nu^d$ for the KR-rearrangement Υ and a reference measure ν^d on $\mathcal{Z}^d$. Let each $\mathcal{Z}$ be equipped with a metric σ and have finite diameter $\|\sigma\| < \infty$. Let $f\colon \mathcal{Z}^d \to \mathbb{R}$ be a Lipschitz function with respect to the product metric σ^d. Let $\mathcal{B} \subseteq \mathcal{Z}^d$ denote a set of bad inputs and define the corresponding set $\mathcal{A} = \Upsilon^{-1}(\mathcal{B}) \subseteq \mathcal{Z}^d$. Let $\widehat{\Upsilon}$ be the unique KR-rearrangement that satisfies $\widehat{\Upsilon}_\sharp\nu^d = \nu^d|\mathcal{A}^c$*

and denote $\widetilde{\Upsilon} = \Upsilon \circ \widehat{\Upsilon}$. *Suppose there exist constants* L_{ij} *such that for all* $x, z \in \mathcal{B}^c$ *with* $x^{[d]\setminus\{i\}} = z^{[d]\setminus\{i\}}$ *it holds*

$$\mathbb{E}_{\tau\sim\nu^{(i,d]}}\Big[\sigma(\widetilde{\Upsilon}^{(i,d]}(\widehat{x}^{[i]},\tau)_j, \widetilde{\Upsilon}^{(i,d]}(\widehat{z}^{[i]},\tau)_j)\Big] \leq L_{ij}\sigma(x_i, z_i) \tag{7.30}$$

where $\widehat{x}^{[i]}$ *and* $\widehat{z}^{[i]}$ *are uniquely defined through* $\widetilde{\Upsilon}^{[i]}(\widehat{x}^{[i]}) = x^{[i]}$ *and* $\widetilde{\Upsilon}^{[i]}(\widehat{z}^{[i]}) = z^{[i]}$. *Then* $\Gamma = \frac{\|\sigma\|}{d} D$ *is a Wasserstein dependency matrix for* $\mu|\mathcal{B}^c$ *with*

$$D_{ij} = \begin{cases} 0 & \text{if } i > j\ , \\ 1 & \text{if } i = j\ , \\ L_{ij} & \text{if } i < j. \end{cases} \tag{7.31}$$

Proof. For arbitrary $z, z' \in \mathcal{Z}^d$ it holds

$$|f(z) - f(z')| \leq \chi_j(f)\sigma(z_j, z'_j), \qquad \forall\, i \in [d] \tag{7.32}$$

and thus, by summing over all indices we get

$$|f(z) - f(z')| \leq \frac{1}{d}\sum_{j\in[d]} \chi_j(f)\sigma(z_j, z'_j) \tag{7.33}$$

Let $x, z \in \mathcal{Z}^d$ with $x^{[d]\setminus\{i\}} = z^{[d]\setminus\{i\}}$ be given for some $i \in [d]$. Recall the action (7.7) of Markov kernels $K^{(i+1)}$ is an expected value with respect to conditional distributions $\mu^{(i,d]}(dy^{(i,d]}|x^{[i]})$.

Because ν^d has no atoms, $\nu^d|\mathcal{A}^c$ also has no atoms. Therefore, there is a unique KR-rearrangement $\widehat{\Upsilon}$ with $\widehat{\Upsilon}_\sharp\nu^d = \nu^d|\mathcal{A}^c$. Then $\widetilde{\Upsilon} = \Upsilon \circ \widehat{\Upsilon}$ is a KR-rearrangement with

$$\widetilde{\Upsilon}_\sharp\nu^d = \mu|\mathcal{B}^c \tag{7.34}$$

by Lemma 7.4 and we have $\widetilde{\Upsilon}(\widehat{x}) = x$. Lemma 7.3 implies

$$\mu^{(i,d]}(dy^{(i,d]}|\mathcal{B}^c, x^{[i]}) = \widetilde{\Upsilon}(\widehat{x}^{[i]}, \cdot)_\sharp\nu^{d-i} \tag{7.35}$$

An analogous expression holds for the distribution conditioned on z. We have therefore found two transport functions pushing the reference measure to the respective conditional distributions. By Lemma 7.7, a coupling of the conditional distributions is then given by

$$P^{[i]}_{x,z} = (\widetilde{\Upsilon}^{(i,d]}(\widehat{x}^{[i]}, \cdot), \widetilde{\Upsilon}^{(i,d]}(\widehat{z}^{[i]}, \cdot))_\sharp\nu^{d-i} \tag{7.36}$$

Using a change of measure we find

$$\begin{aligned}
&K^{(i+1)}f(x) - K^{(i+1)}f(z) \\
&\quad= \int P^{[i]}_{x,z}(du^{(i,d]}, dv^{(i,d]})\Big(f(x^{[i]}u^{(i,d]}) - f(z^{[i]}v^{(i,d]})\Big) && (7.37)\\
&\quad= \int \Big(f(x^{[i]}\widetilde{\Upsilon}^{(i,d]}(\widehat{x}^{[i]},\tau)) - f(z^{[i]}\widetilde{\Upsilon}^{(i,d]}(\widehat{z}^{[i]},\tau))\Big)\nu^{d-i}(\tau) && (7.38)\\
&\quad\leq \frac{\chi_i(f)}{d}\sigma(x_i, z_i) + \sum_{j\in(i,d]} \frac{\chi_j(f)}{d}\int \sigma\Big(\widetilde{\Upsilon}^{(i,d]}(\widehat{x}^{[i]},\tau)_j, \widetilde{\Upsilon}^{(i,d]}(\widehat{z}^{[i]},\tau)_j\Big)\nu^{d-i}(\tau) && (7.39)\\
&\quad\leq \frac{\chi_i(f)}{d}\sigma(x_i, z_i) + \sum_{j\in(i,d]} \frac{\chi_j(f)}{d} L_{ij}\sigma(x_i, z_i) && (7.40)
\end{aligned}$$

which shows

$$\chi_i(K^{(i+1)}f) \leq \frac{1}{d}\Big(\chi_i(f) + \sum_{j\in(i,d]} L_{ij}\chi_j(f)\Big) \tag{7.41}$$

for good inputs. We have thus found a Wasserstein matrix $V^{(i+1)}$ for $K^{(i+1)}$ with entries

$$V_{ij}^{(i+1)} = \begin{cases} 0 & \text{if } i > j \\ d^{-1} & \text{if } i = j \\ d^{-1}L_{ij} & \text{if } i < j \end{cases} \tag{7.42}$$

in row i which shows the assertion. □

We remark that Γ indeed distills the dependency structure of μ. To illustrate this, let $\mu^{\{i\}}$ be independent from $\mu^{\{j\}}$ for some $i, j \in [d]$. Then conditioning on a different value of $\mu^{\{i\}}$ does not change the distribution $\mu^{\{j\}}$. Thus,

$$\widetilde{\Upsilon}^{(i,d]}(\overline{x}^{[i]}, \tau)_j = \widetilde{\Upsilon}^{(i,d]}(\overline{z}^{[i]}, \tau)_j, \qquad \forall\, \tau \in \mathcal{Z}^{(i,d]}, \quad \overline{x}^{[d]\setminus\{i\}} = \overline{z}^{[d]\setminus\{i\}}, \quad j \in (i,d] \tag{7.43}$$

and the choice $L_{ij} = 0$ satisfies (7.30). It directly follows that $\Gamma_{ij} = 0$. The following theorem is the main result of this chapter.

Theorem 7.9 (PAC-Bayesian risk certificate for structured prediction) *Fix $\delta \in (0, \exp(-e^{-1}))$, let μ be a data distribution on $\mathcal{Z}^d$ with $\Upsilon_\sharp \nu^d = \mu$ the Knothe-Rosenblatt rearrangement for a reference measure ν^d on $\mathcal{Z}^d$ and fix a measurable set $\mathcal{B} \subseteq \mathcal{Z}^d$ of bad inputs with $\mu(\mathcal{B}) \leq \xi$. Fix a PAC-Bayes prior π on a hypothesis class $\mathcal{H}$ of functions $\phi\colon \mathcal{X}^d \to \mathcal{Y}^d$ and a loss function ℓ which assumes values in $[0,1]$. Define the oscillation vector $\widetilde{\chi}$ by*

$$\widetilde{\chi}_i = \sup_{h \in \mathcal{H}} \chi_i\Big(L(h,\cdot)\Big|_{\mathcal{B}^c}\Big), \qquad i \in [d] \tag{7.44}$$

where $L(h,\cdot)\Big|_{\mathcal{B}^c}$ denotes the restriction of $L(h,\cdot)$ to $\mathcal{Z}^d \setminus \mathcal{B}$. Suppose all oscillations $\widetilde{\chi}_i$ are finite, suppose the condition (7.30) is satisfied and denote by D the matrix with entries (7.31). Then, with probability at least $1-\delta$ over realizations of a training set $\mathcal{D}_m = (Z^{(k)})_{k=1}^m$ drawn from $(\mu|\mathcal{B}^c)^m$ it holds for all PAC-Bayes posteriors ρ on $\mathcal{H}$ that

$$\mathcal{R}(\rho) \leq \mathcal{R}_m(\rho, \mathcal{D}_m) + 2\frac{\|\sigma\|}{d}\|D\widetilde{\chi}\|_2 \sqrt{\frac{\log\frac{1}{\delta} + \mathrm{KL}[\rho:\pi]}{2m}} + \xi. \tag{7.45}$$

Proof. For any hypothesis $h \in \mathcal{H}$, we have

$$\begin{aligned}
\mathcal{R}(h) - \mathcal{R}_m(h, \mathcal{D}_m) &= \mathbb{E}_{Z\sim\mu}[L(h,Z) - \mathcal{R}_m(h,\mathcal{D}_m)] && \text{(7.46a)}\\
&= \mathbb{E}_{Z\sim\mu}\left[\Big(L(h,Z) - \mathcal{R}_m(h,\mathcal{D}_m)\Big)\mathbb{1}\{Z \notin \mathcal{B}\}\right] \\
&\qquad + \mathbb{E}_{Z\sim\mu}\left[\Big(L(h,Z) - \mathcal{R}_m(h,\mathcal{D}_m)\Big)\mathbb{1}\{Z \in \mathcal{B}\}\right] && \text{(7.46b)}\\
&\leq \mathbb{E}_{Z\sim\mu}\left[\Big(L(h,Z) - \mathcal{R}_m(h,\mathcal{D}_m)\Big)\mathbb{1}\{Z \notin \mathcal{B}\}\right] + \xi && \text{(7.46c)}\\
&\leq \mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - \mathcal{R}_m(h,\mathcal{D}_m) + \xi && \text{(7.46d)}
\end{aligned}$$

where in (7.46c) we have used that pointwise loss is in $[0,1]$. Note that the underlying distribution of the risk $\mathcal{R}(h)$ is μ, while $\mathcal{D}_m$ are drawn from $\mu|\mathcal{B}^c$. The above inequality reconciles this such that a concentration argument for the conditional distribution becomes applicable. For any PAC-Bayes posterior distribution ρ and any $\beta > 0$, this implies

$$\begin{aligned}
\mathcal{R}(\rho) - \mathcal{R}_m(\rho, \mathcal{D}_m) &= \mathbb{E}_{h\sim\rho}\mathbb{E}_{Z\sim\mu}[L(h,Z) - \mathcal{R}_m(h,\mathcal{D}_m)] && \text{(7.47a)}\\
&\leq \mathbb{E}_{h\sim\rho}\Big[\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - \mathcal{R}_m(h,\mathcal{D}_m)\Big] + \xi && \text{(7.47b)}\\
&= \frac{1}{\beta}\mathbb{E}_{h\sim\rho}\Big[\beta(\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - \mathcal{R}_m(h,\mathcal{D}_m))\Big] + \xi && \text{(7.47c)}\\
&\leq \frac{1}{\beta}\log \mathbb{E}_{h\sim\pi}\Big[\exp\Big(\beta(\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - \mathcal{R}_m(h,\mathcal{D}_m))\Big)\Big] \\
&\qquad + \frac{1}{\beta}\mathrm{KL}[\rho:\pi] + \xi && \text{(7.47d)}
\end{aligned}$$

by Donsker and Varadhan's variational formula (Lemma 2.15). Focusing on the first term,

we find

$$\exp\Big(\beta(\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - \mathcal{R}_m(h,\mathcal{D}_m))\Big)$$
$$= \exp\Big(\frac{\beta}{m}\sum_{k\in[m]}\Big(\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - L(h,Z^{(k)})\Big)\Big) \tag{7.48a}$$
$$= \prod_{k\in[m]}\exp\Big(\frac{\beta}{m}\Big(\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - L(h,Z^{(k)})\Big)\Big) \tag{7.48b}$$

Each structured datum $Z^{(k)}$ is drawn independently from $\mu|\mathcal{B}^c$. By Proposition 7.8 there exists a Wasserstein dependency matrix $\Gamma = \frac{\|\sigma\|}{d}D$ for $\mu|\mathcal{B}^c$ where D has entries (7.31). Then it holds

$$\mathbb{E}_{\mathcal{D}_m\sim(\mu|\mathcal{B}^c)^m}\prod_{k\in[m]}\exp\Big(\frac{\beta}{m}\Big(\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - L(h,Z^{(k)})\Big)\Big)$$
$$= \prod_{k\in[m]}\mathbb{E}_{Z^{(k)}\sim(\mu|\mathcal{B}^c)}\exp\Big(\frac{\beta}{m}\Big(\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - L(h,Z^{(k)})\Big)\Big) \tag{7.49a}$$
$$= \prod_{k\in[m]}\mathbb{E}_{Z^{(k)}\sim\mu|\mathcal{B}^c}\Big[\exp\Big(\frac{\beta}{m}\Big(\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - L(h,Z^{(k)})\Big)\Big)\Big] \tag{7.49b}$$
$$\leq \prod_{k\in[m]}\exp\Big(\frac{\beta^2}{8m^2}\|\Gamma\chi(\widetilde{L}(h,\cdot))\|_2^2\Big) \text{ by Theorem 7.6} \tag{7.49c}$$
$$= \exp\Big(\frac{\beta^2}{8m}\|\Gamma\chi(\widetilde{L}(h,\cdot))\|_2^2\Big) \leq \exp\Big(\frac{\beta^2}{8m}\|\Gamma\widetilde{\chi}\|_2^2\Big). \tag{7.49d}$$

Define the shorthand

$$U = \mathbb{E}_{\mathcal{D}_m\sim(\mu|\mathcal{B}^c)^m}\Big[\exp\Big(\beta(\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - \mathcal{R}_m(h,\mathcal{D}_m))\Big)\Big]. \tag{7.50}$$

By Markov's inequality it holds

$$\mathbb{P}_{\mathcal{D}_m\sim(\mu|\mathcal{B}^c)^m}\Big[\exp\Big(\beta(\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - \mathcal{R}_m(h,\mathcal{D}_m))\Big) \geq \frac{1}{\delta}U\Big] \leq \delta \tag{7.51}$$

and combining this with (7.49) we have

$$\exp\Big(\beta(\mathbb{E}_{Z\sim\mu|\mathcal{B}^c}[L(h,Z)] - \mathcal{R}_m(h,\mathcal{D}_m))\Big) \leq \frac{1}{\delta}\exp\Big(\frac{\beta^2}{8m}\|\Gamma\widetilde{\chi}\|_2^2\Big) \tag{7.52}$$

with probability at least $1-\delta$ over the sample. Using (7.47) we find

$$\mathcal{R}(\rho) - \mathcal{R}_m(\rho,\mathcal{D}_m) \leq \frac{1}{\beta}\Big(\log\mathbb{E}_{h\sim\pi}\Big[\frac{1}{\delta}\exp\Big(\frac{\beta^2}{8m}\|\Gamma\widetilde{\chi}\|_2^2\Big)\Big] + \mathrm{KL}[\rho:\pi]\Big) + \xi \tag{7.53a}$$
$$= \frac{\beta}{8m}\|\Gamma\widetilde{\chi}\|_2^2 + \frac{1}{\beta}\Big(\log\frac{1}{\delta} + \mathrm{KL}[\rho:\pi]\Big) + \xi. \tag{7.53b}$$

Ideally, we would minimize the right hand side with respect to β. However, this would mean to have β depend on ρ and we thus would not have a uniform bound for all posterior distributions. Instead, [126] approaches the problem by defining a sequence of constant $(\delta_j, \beta_j)_{j\in\mathbb{N}_0}$ and bounding the probability that the bound does not hold for any sequence element. Since in the opposite (high-probability) case, the bound holds for all sequence elements, an optimal one can subsequently be chosen dependent on the posterior. For all $j \in \mathbb{N}_0$, define

$$\delta_j = \delta 2^{-(j+1)}, \qquad \beta_j = 2^j \sqrt{\frac{8m \log \frac{1}{\delta}}{\|\Gamma\widetilde{\chi}\|_2^2}} \tag{7.54}$$

which are independent of ρ. Now consider the event E_j that

$$\exp\left(\beta_j(\mathbb{E}_{X\sim\mu|\mathcal{B}^c}[\ell(h,X)] - \mathcal{R}_m(h,\mathcal{D}_m))\right) \geq \frac{1}{\delta_j}\exp\left(\frac{\beta_j^2}{8m}\|\Gamma\widetilde{\chi}\|_2^2\right). \tag{7.55}$$

By the above argument leading up to (7.52), the probability for E_j under a random sample of the conditioned data distribution $\mu|\mathcal{B}^c$ is at most δ_j. Therefore, the probability that any E_j occurs is bounded by

$$\mathbb{P}\Big(\bigcup_{j\in\mathbb{N}_0} E_j\Big) \leq \sum_{j\in\mathbb{N}_0} \mathbb{P}(E_j) \leq \sum_{j\in\mathbb{N}_0} \delta_j = \delta. \tag{7.56}$$

Thus, for all posteriors ρ with probability at least $1-\delta$ none of the events (7.55) occurs. We may therefore select an index j dependent on ρ to obtain a sharper risk certificate which still holds with probability at least $1-\delta$ over the sample conditioned on the good set. For a fixed posterior ρ, the optimizer of (7.53b) would be

$$\beta^* = \frac{1}{\|\Gamma\widetilde{\chi}\|_2}\sqrt{8m(\log\frac{1}{\delta} + \mathrm{KL}[\rho:\pi])}. \tag{7.57}$$

Equating this to (7.55) and rounding down to the nearest integer gives

$$j^* = \left\lfloor \frac{1}{2}\log_2\left(1 + \frac{\mathrm{KL}[\rho:\pi]}{\log\frac{1}{\delta}}\right)\right\rfloor. \tag{7.58}$$

Denote this number before rounding by r, i.e. $j^* = \lfloor r \rfloor$. For any real number r it holds $r - 1 \leq \lfloor r \rfloor \leq r$. Therefore

$$\frac{1}{2}\sqrt{1 + \frac{\mathrm{KL}[\rho:\pi]}{\log\frac{1}{\delta}}} = 2^{r-1} \leq 2^{j^*} \leq 2^r = \sqrt{1 + \frac{\mathrm{KL}[\rho:\pi]}{\log\frac{1}{\delta}}} \tag{7.59}$$

which gives the following bounds on u_{j^*}

$$\frac{1}{2}\sqrt{\frac{8m(\log\frac{1}{\delta} + \mathrm{KL}[\rho:\pi])}{\|\Gamma\widetilde{\chi}\|_2^2}} \leq u_{j^*} \leq \sqrt{\frac{8m(\log\frac{1}{\delta} + \mathrm{KL}[\rho:\pi])}{\|\Gamma\widetilde{\chi}\|_2^2}}. \tag{7.60}$$

Likewise, we bound

$$\begin{aligned}
\mathrm{KL}[\rho : \pi] + \log \frac{1}{\delta_{j^*}} &= \mathrm{KL}[\rho : \pi] + \log \frac{2}{\delta} + j^* \log 2 && \text{(7.61a)}\\
&\leq \mathrm{KL}[\rho : \pi] + \log \frac{2}{\delta} + \frac{\log 2}{2} \log_2 \left(1 + \frac{\mathrm{KL}[\rho : \pi]}{\log \frac{1}{\delta}}\right) - \log 2 && \text{(7.61b)}\\
&= \mathrm{KL}[\rho : \pi] + \log \frac{1}{\delta} + \frac{1}{2} \log \left(1 + \frac{\mathrm{KL}[\rho : \pi]}{\log \frac{1}{\delta}}\right) && \text{(7.61c)}\\
&= \mathrm{KL}[\rho : \pi] + \log \frac{1}{\delta} + \frac{1}{2} \log \Big(\log \frac{1}{\delta} + \mathrm{KL}[\rho : \pi]\Big) - \frac{1}{2} \log \log \frac{1}{\delta}. && \text{(7.61d)}
\end{aligned}$$

The assumption $\delta \leq \exp(-e^{-1})$ yields $-\log \log \frac{1}{\delta} \leq 1$ and because $x + 1 \leq \exp(x)$ for all $x \in \mathbb{R}$, we find

$$\begin{aligned}
\mathrm{KL}[\rho : \pi] + \log \frac{1}{\delta_{j^*}} &\leq \mathrm{KL}[\rho : \pi] + \log \frac{1}{\delta} + \frac{1}{2}\Big(\log \Big(\log \frac{1}{\delta} + \mathrm{KL}[\rho : \pi]\Big) + 1\Big) && \text{(7.62a)}\\
&\leq \mathrm{KL}[\rho : \pi] + \log \frac{1}{\delta} + \frac{1}{2}\Big(\log \frac{1}{\delta} + \mathrm{KL}[\rho : \pi]\Big) && \text{(7.62b)}\\
&= \frac{3}{2}\Big(\log \frac{1}{\delta} + \mathrm{KL}[\rho : \pi]\Big). && \text{(7.62c)}
\end{aligned}$$

We can now use the bounds (7.62c) and (7.60) in (7.53b) to bound the expected generalization error

$$\begin{aligned}
\mathcal{R}(\rho) - \mathcal{R}_m(\rho, \mathcal{D}_m) &\leq \frac{u_{j^*}}{8m} \|\Gamma \widetilde{\chi}\|_2^2 + \frac{1}{u_{j^*}} \Big(\log \frac{1}{\delta_{j^*}} + \mathrm{KL}[\rho : \pi]\Big) + \xi && \text{(7.63a)}\\
&\leq \frac{u_{j^*}}{8m} \|\Gamma \widetilde{\chi}\|_2^2 + \frac{3}{2u_{j^*}} \Big(\log \frac{1}{\delta} + \mathrm{KL}[\rho : \pi]\Big) + \xi && \text{(7.63b)}\\
&\leq \frac{1}{2} \|\Gamma \widetilde{\chi}\|_2 \sqrt{\frac{\log \frac{1}{\delta} + \mathrm{KL}[\rho : \pi]}{2m}}\\
&\quad + \frac{3}{2} \|\Gamma \widetilde{\chi}\|_2 \sqrt{\frac{\log \frac{1}{\delta} + \mathrm{KL}[\rho : \pi]}{2m}} + \xi && \text{(7.63c)}\\
&= 2 \|\Gamma \widetilde{\chi}\|_2 \sqrt{\frac{\log \frac{1}{\delta} + \mathrm{KL}[\rho : \pi]}{2m}} + \xi && \text{(7.63d)}
\end{aligned}$$

Note that β^* would attain the optimal value

$$\mathcal{R}(\rho) - \mathcal{R}_m(\rho, \mathcal{D}_m) \leq \|\Gamma \widetilde{\chi}\|_2 \sqrt{\frac{\mathrm{KL}[\rho : \pi] + \log \frac{1}{\delta}}{2m}} + \xi \tag{7.64}$$

which only differs from the above uniform bound by a factor of two. Finally, recall $\Gamma = \frac{\|\sigma\|}{d} D$ where D has entries (7.31). □

Note that the generalization gap on the right hand side of (7.45) decays with d. This accounts for generalization from a *single* example. In fact, if only $m = 1$ structured

example is available, but $d \gg 1$, Theorem 7.9 still certifies risk. This effect can however be negated by the norm $\|D\widetilde{\chi}\|$. If structured data contain strong global dependence, then $\|D\widetilde{\chi}\|$ will not be bounded independently of d and thus, in the worst case of $\|D\widetilde{\chi}\| \in \mathcal{O}(d)$ the assertion is no stronger than PAC-Bayesian bounds for unstructured data. The same point was observed in [126].

The measure of the bad set under the data distribution μ is assumed to be bounded by ξ. This is to account for a small number of data which contain strong dependence. In order to prevent these bad data from dominating D, thereby negating the decay of the bound in d as described above, it is preferable to exclude them from the sample, reduce the sample size m and pay the penalty ξ in (7.45).

To prove Theorem 7.9, we broadly follow the argument put forward in [126]. This augments typical PAC-Bayesian constructions in the literature by the inclusion of a set of bad inputs. We first reconcile the data being conditioned on $\mathcal{B}^c$ with risk certification for unconditioned data, leading to the addition of ξ on the right hand side of (7.45). The model complexity term $\mathrm{KL}[\rho : \pi]$ is due to Donsker and Varadhand's variational formula. Subsequently, the moment generating function bound of Theorem 7.6 is instantiated through the Wasserstein dependency matrix constructed in Proposition 7.8. Markov's inequality then gives a pointwise risk bound for fixed value of a free parameter. In order to optimize this parameter, the bound is made uniform on a discrete set of values through a union bound.

In Theorem 7.9, we combine the PAC-Bayesian construction of [126] with the more general concentration of measure theory of [111]. Crucially, concentration of measure results used in [126] are predicated on the assumption of data generated by a Markov random field [214, 110]. Our work is more flexible in two major ways.

(1) Our assumption on the data-generating distribution is likely more representative of real-world data as measure transport models have repeatedly been shown to yield convincing data generators.

(2) Markov random fields are difficult to handle computationally because inference in general Markov random fields is NP-hard [214] such that one is forced to learn based on approximate inference procedures [102, 61, 204].

Our work also allows for more general metrics σ as opposed to the singular choice of Hamming norm required in [126]. Additionally, the key results of [126] are constructed to ensure all data drawn from the unconditioned distribution μ are in the good set. This reduces the probability of correctness $1 - \delta$ by $m\xi$. Instead, we assume data drawn from $\mu|\mathcal{B}^c$, effectively reducing the number of available samples by a factor of $1 - \xi$, but keeping the probability of correctness high. This allows the set of bad inputs to be used more effectively as a computational tool in Section 7.3.

Comparing the dependency of (7.45) on d with the respective result in [126], it first appears as though our bound decays with a faster rate (d instead of $\sqrt{d}$). However, this will not typically be the case in practice because $\|D\widetilde{\chi}\|_2$ grows with rate $\sqrt{d}$ in most

situations. To see this, consider the case of local dependency in the sense that

$$L_{ij} = \begin{cases} 1, \text{ if } j \in \mathcal{N}_i \ , \\ 0, \text{ else} \end{cases} \tag{7.65}$$

for local neighborhoods $\mathcal{N}_i \subseteq [d]$ which contain a fixed number of c elements and let $\widetilde{\chi}_i = \alpha$ for all $i \in [d]$ and some constant value $\alpha > 0$. Then

$$\|D\widetilde{\chi}\|_2 = \sqrt{\sum_{i \in [d]} \Big(\alpha |\mathcal{N}_i|\Big)^2} = c\alpha\sqrt{d}. \tag{7.66}$$

Clearly, if dependence is localized and the oscillations $\widetilde{\chi}$ do not decay in d, then $\|D\widetilde{\chi}\|_2$ contains a factor that grows with rate $\sqrt{d}$, leading to the same asymptotic rate of (7.45) already observed in [126].

Note that [126] additionally allows for a set of bad hypotheses $\mathcal{B}_{\mathcal{F}} \subseteq \mathcal{H}$ which do not conform to stability assumptions. In our construction, this means restricting the bound (7.44) to oscillations on the set of good hypotheses. We omit this extension for clarity of exposition, but do not expect it to necessitate major changes to the presented proofs.

Further, [126] considers a large number of applicable orderings for random variables by introducing a filtration of their index set. This notion is not easily compatible with our assumption of KR-rearrangement, because triangularity of transport depends on the order of variables.

7.3 Bounding the Bad Set

With regard to numerical risk certificates, a key technical aspect of Section 7.2 concerns the quantities L_{ij} in (7.30). Here, we propose a way to use the set of bad inputs as a computational tool to this end. Suppose we assign arbitrary fixed values to L_{ij} and subsequently *define* $\mathcal{B} \subseteq \mathcal{Z}^d$ as the set of inputs on which the condition (7.30) fails. Then we have fulfilled the prerequisites of Proposition 7.8 by construction and are left with bounding $\mu(\mathcal{B})$. Note that

$$\mu(\mathcal{B}) = \mathbb{P}_{Z \sim \mu}(Z \in \mathcal{B}) = \mathbb{E}_{Z \sim \mu}[\mathbb{1}\{Z \in \mathcal{B}\}] \tag{7.67}$$

and the indicator function $\mathbb{1}$ assumes values in the bounded set $\{0, 1\}$. Therefore, Hoeffding's inequality gives the following.

Proposition 7.10 (Upper bound on the bad set) *Let μ be a data distribution and let $\widetilde{\mathcal{D}}_n \sim \mu^n$ be a sample of size n. Fix an error probability $\epsilon \in (0, 1)$. Then*

$$\mu(\mathcal{B}) \leq \frac{1}{n} \sum_{Z \in \widetilde{\mathcal{D}}_n} \mathbb{1}\{Z \in \mathcal{B}\} + \sqrt{\frac{1}{2n} \log \frac{2}{\epsilon}} \tag{7.68}$$

with probability at least $1 - \epsilon$ over the sample.

Checking the condition $Z \in \mathcal{B}$ requires evaluating (7.30) which comes down to finding a Lipschitz constant for a one-dimensional function. If this is computationally feasible for the given data model, then the concentration argument (7.68) can be used to bound $\mu(\mathcal{B})$ with high probability. Because (7.68) decays only in the number of structured examples $\mathcal{O}(\sqrt{n})$, it can not be used to show generalization from a single structured example. However, PAC-Bayesian risk certificates are typically dominated by the KL complexity term in (7.45) which decays with the size of structured examples as well. Thus, Proposition 7.10 should still be useful in practice.

Note that Proposition 7.10 makes a pointwise statement about a fixed value of L_{ij} which has limited utility for learning L_{ij} from data. To remedy this problem, we can first define a discrete set $\mathcal{L} = (L^{(k)})_{k\in[l]}$ of candidate matrices and select error probabilities ϵ_k for each of the events

$$\mu(\mathcal{B}(L^{(k)})) \geq \frac{1}{n} \sum_{Z \in \widetilde{\mathcal{D}}_n} \mathbb{1}\{Z \in \mathcal{B}(L^{(k)})\} + \sqrt{\frac{1}{2n} \log \frac{2}{\epsilon_k}} \tag{7.69}$$

such that $\sum_{k\in[l]} \epsilon_k = \epsilon$. Then, by a union bound with probability at least $1 - \epsilon$ over the sample none of the events (7.69) occurs. We have thus constructed a uniform bound over the set of candidate matrices which allows us to select the one which minimizes the generalization gap in (7.45).

In order to make this strategy most effictive, domain knowledge on the application at hand should be applied when constructing candidate matrices and assigning error probabilities. For instance, the limited empirical findings of [45] on ImageNet [53] indicate that the majority of natural images contain mostly local signal. In image segmentation, this is conducive to concentration, because it can lead to many small values in an optimal Wasserstein dependency matrix. In particular, if dependency decays with distance in the image domain, one should select configurations $L^{(k)}$ in which $L^{(k)}_{ij}$ is small if i is distant from j in the image domain and allowed to assume larger values if i is close to j in the image domain.

We give an *intuitive interpretation* of the relationship between Proposition 7.10 and Theorem 7.9 as follows. Suppose the majority of samples from a structured data distribution contain mostly local signal. The locality of signal in samples indicates weak global dependence of random variables which in turn manifests in small entries of a Wasserstein dependency matrix. However, a small number of bad data may contain only weak local signal. For instance, an image in which every pixel has the same value does not give more information to a learner if it is doubled in size. Even worse, a small (but not null) set of bad data will dominate the Wasserstein dependency matrix and prevent generalization that scales with d. Proposition 7.10 thus estimates an upper bound on the likelihood of bad data under μ which are then excluded from the concentration argument underlying the bound (7.45).

8 Conclusion and Outlook

This thesis has studied structured prediction, the problem of jointly predicting realizations of multiple coupled random variables. It was broadly organized into a first part, focusing on *geometric* aspects, and a second part, focusing on *statistical* aspects.

Chapter 4 layed a geometric foundation by studying the set of factorizing discrete probability distributions as an embedded submanifold of a meta-simplex which contains *all* joint distributions of multiple discrete random variables. This was put into the perspective of game theory and information geometry and we ultimately found a correspondence between assignment flows on the embedded submanifold and replicator dynamics with structured payoff functions on the meta simplex. From a game theoretical viewpoint, the proposed embedding framework provides a robust mathematical toolset for modeling complex population interactions. It simplifies analysis by formally reducing the complex multi-population case to a single-population one.

Building on this geometric understanding of factorizing distributions relative to general joint distributions, two generative models of discrete data were developed in Chapter 5. The first, described in Section 5.2, has aimed to approximate a given energy-based model. Applications are abundant in statistical physics and a core technical question concerned differentiable estimation of model entropy. The second generative model, described in Section 5.3, has aimed to approximate joint distributions of discrete random variables which are only accessible through a dataset of samples. This has lead to the construction of continuous normalizing assignment flows, which we trained by employing a Riemannian flow-matching approach. In both cases considered in Chapter 5, the underlying idea was to parameterize joint distributions $p \in \mathcal{S}_N$ by distributions $\nu \in \mathcal{P}(\mathcal{W})$ supported on the much lower-dimensional assignment manifold.

Turning to statistical learning, a self-certified image classification approach was proposed in Chapter 6. The method was built on a linearization of assignment flows and achieved normal distribution of classification logits with tractable moments. This was leveraged to gain favorable computational properties within the PAC-Bayesian risk certification paradigm, by allowing to bound expected empirical risk under the PAC-Bayesian posterior distribution efficiently and with high probability.

In Chapter 7 the statistical part of the thesis has culminated in a novel PAC-Bayesian generalization bound for structured prediction. It was argued that a model for the data-generating distribution is required in this case and the choice of Knothe-Rosenblatt rearrangement was proposed. This shape of measure transport model allowed to distill quantities which characterize dependencies in the joint distribution of data, called a Wasserstein dependency matrix. In turn, state of the art concentration of measure theory was invoked, ultimately allowing a PAC-Bayesian construction.

8.1 Future Work

The variety of topics and literature touched in this thesis illustrates that structured prediction is a complex problem domain, stimulating ongoing research on both theory and applications. Here, we list open questions and future directions which appear as natural next steps in light of the presented work.

8.1.1 Assignment Flows

In [183], assignment flows are characterized as critical points of an action functional within a geometric formalism of mechanics. This generalizes an analogous characterization, previously suggested for replicator dynamics [165] on single populations. The embedding theorem 4.5 presented in Chapter 4 allows to formally reduce assignment flows to single-population replicator dynamics, incurring high dimension and structured payoff. A natural question is whether the characterization proposed in [183] is equivalent to the one of [165] under this embedding.

A generalized perspective on assignment flows was proposed by [187]. The authors study a dynamical system on a product of density matrix manifolds called *Quantum State Assignment Flow*. Although density matrices can represent entangled states and constitute a strict generalization of discrete probability measures, the underlying information geometric framework is broadly analogous. This suggests the possibility of generalizing the embedding results of Chapter 4 to the density matrix domain.

8.1.2 Generative Models

The approximation of energy-based models by stochastic assignment flows proposed in Section 5.2 builds on a differentiable estimator for model entropy. However, we have employed a simple estimator, which is limited by slow convergence for distributions with large support. A natural direction of future work is to construct similar differentiable approximations for more advanced entropy estimators.

Flow matching and other methods of learning generative models from discrete data are currently very active areas of research. An approach which is closely related to the one of Section 5.3 was concurrently proposed by [196]. The authors model conditional probability paths of measures $\nu(t) \in \mathcal{P}(\mathcal{W})$ as e-geodesic paths in the exponential family of Dirichlet

distributions on each simplex. They argue that, by not transporting all probability mass of the reference measure to a single region $A_\gamma \subseteq \mathcal{W}$ in finite time, decisionmaking is less compressed into a short time interval, leading to more effective use of model capacity. This reasoning stimulates the idea of generalizing the approach of Section 5.3 to match paths of conditional measures on $T_0\mathcal{W}$ which escape to infinity for $t \to \infty$ instead of concentrating on points $U^{(\gamma)}$ (defined in (5.49)) after finite time.

8.1.3 Statistical Learning Theory and Deep Learning

We have employed a differentiable surrogate loss in place of 0/1 loss for training PAC-Bayesian posteriors in Chapter 6. It was pointed out by [48] that, while 0/1 loss has vanishing gradient and thus provides no useful training signal, *expected* empirical risk relative to 0/1 loss does not need to have the same property. In the case of binary classification and normal distribution of logits, the authors derive a closed form of expected empirical risk, demonstrating non-vanishing gradients. This technique was developed further in [47], providing unbiased estimators for $c > 2$. Since the stochastic LAF classifier constructed in Chapter 6 produces classification logits following a normal distribution for each datum, implementing direct minimization of expected empirical risk without surrogate loss is a natural extension. In addition, direct numerical optimization of the PAC-Bayes-kl inequality, without a relaxation like (6.36) could further improve the sharpness of risk certificates.

The PAC-Bayesian generalization bound for structured prediction proposed in Chapter 7 relies on the construction of a Wasserstein dependency matrix. In order to evaluate such a bound numerically, methods of computing the Lipschitz constants L_{ij}, related to the transport map through (7.30), need to be developed. From an application perspective, it would also be beneficial to generalize the proposed approach to other measure transport methods. This would make it easier to leverage existing foundation models [27] trained on a broad range of data.

The assumption that data is generated by Knothe-Rosenblatt rearrangement is weak, but to get access to (an approximation of) this transport map in practice requires learning a generative model from data. Thus, from a learning theoretical standpoint, the main limitation of our PAC-Bayesian approach to structured prediction is the lack of generalization theory for generative models. For normalizing flows, an asymptotic approach was proposed in the recent work [14]. Another recent line of work has developed non-vacuous PAC-Bayesian bounds for the generalization of variational autoencoders [140], generative adversarial networks [139] and diffusion models [141]. These approaches are promising but have not yet been scaled to large datasets. If high-probability generalization bounds for foundation models become available, a further open question is how the slack of these bounds affects the tightness of bounds on downstream discriminative task risk through our construction.

The main limitation of current PAC-Bayesian approaches is their strong reliance on data-dependent priors to achieve tightness. This can also be seen in Chapter 6 where

expected empirical risk of the prior on test data is already close to the performance of the posterior. The detailed empirical work [161] examined the role of the prior in PAC-Bayesian learning. The authors find that, although there is a dependency on the dataset, the tightest certificates are achieved by using a substantial amount – even the majority – of all available data for prior learning. This indicates that in order to achieve a tight certificate which can inform the posterior learning process in a principled way, we need to have access to a prior that already generalizes well to unseen data. In comparison with the validation set certificates of [104], all data can be used to learn PAC-Bayesian posteriors, which underlies the superior explanatory power of the PAC-Bayesian approach. However, strong reliance on data-dependent priors for tightness still limits the extent to which PAC-Bayesian bounds can serve as a principled basis for deep *learning* beyond validation.

Bibliography

[1] Josh Achiam et al. *GPT-4 technical report*. 2023. arXiv: 2303.08774.

[2] Christoph Aistleitner et al. "On functions of bounded variation". In: *Mathematical Proceedings of the Cambridge Philosophical Society*. Vol. 162. 3. Cambridge University Press. 2017, pp. 405–418.

[3] Pierre Alquier. *User-friendly introduction to PAC-Bayes bounds*. 2023. arXiv: 2110.11216 [stat.ML].

[4] Pierre Alquier and Benjamin Guedj. "Simpler PAC-Bayesian bounds for hostile data". In: *Machine Learning* 107.5 (2018), pp. 887–902.

[5] Felipe Alvarez, Jérôme Bolte, and Olivier Brahic. "Hessian Riemannian gradient flows in convex programming". In: *SIAM journal on control and optimization* 43.2 (2004), pp. 477–501.

[6] Shun-Ichi Amari. *Information Geometry and Its Applications*. Applied Mathematical Sciences ; 194. Tokyo: Springer, 2016. ISBN: 978-4-431-55978-8. DOI: 10.1007/978-4-431-55978-8.

[7] Shun-ichi Amari and Hiroshi Nagaoka. *Methods of Information Geometry*. Vol. 191. Amer. Math. Soc. and Oxford Univ. Press, 2000.

[8] Lynton Ardizzone et al. *Guided image generation with conditional invertible neural networks*. 2019. arXiv: 1907.02392.

[9] Walter Edwin Arnoldi. "The principle of minimized iterations in the solution of the matrix eigenvalue problem". In: *Quarterly of applied mathematics* 9.1 (1951), pp. 17–29.

[10] J. Ashkin and E. Teller. "Statistics of Two-Dimensional Lattices with Four Components". In: *Phys. Rev.* 64 (5-6 Sept. 1943), pp. 178–184. DOI: 10.1103/PhysRev.64.178.

[11] Freddie Åström et al. "Image Labeling by Assignment". In: *Journal of Mathematical Imaging and Vision* 58.2 (Jan. 2017), pp. 211–238.

[12] Nihat Ay et al. *Information Geometry*. Vol. 64. A Series of Modern Surveys in Mathematics. Cham: Springer, 2017. ISBN: 978-3-319-56478-4. DOI: 10.1007/978-3-319-56478-4.

[13] Ricardo Baptista, Youssef Marzouk, and Olivier Zahm. *On the representation and learning of monotone triangular transport maps*. arXiv:2009.10303 [cs, math, stat]. July 2022.

[14] Ricardo Baptista et al. *An Approximation Theory Framework for Measure-Transport Sampling Algorithms*. 2023. arXiv: 2302.13965.

[15] R.J. Baxter. *Exactly Solved Models in Statistical Mechanics*. Academic Press, 1982.

[16] A.G. Baydin et al. "Automatic Differentiation in Machine Learning: a Survey". In: *J. Machine Learning Research* 18 (2018), pp. 1–43.

[17] Jenna Bednar and Scott Page. "Can Game(s) Theory Explain Culture?: The Emergence of Cultural Behavior Within Multiple Games". In: *Rationality and Society* 19.1 (2007), pp. 65–97. DOI: 10.1177/1043463107075108.

[18] H. Ben-Hamu et al. "Matching Normalizing Flows and Probability Paths on Manifolds". In: *International Conference on Machine Learning*. 2022.

[19] I. Bengtsson and K. Zyczkowski. *Geometry of Quantum States: An Introduction to Quantum Entanglement*. 2nd. Cambridge University Press, 2017.

[20] Felix Biggs. *A Note on the Efficient Evaluation of PAC-Bayes Bounds*. 2022. arXiv: 2209.05188.

[21] Felix Biggs and Benjamin Guedj. "Non-Vacuous Generalisation Bounds for Shallow Neural Networks". In: *Proceedings of the 39th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri et al. Vol. 162. Proceedings of Machine Learning Research. PMLR, July 2022, pp. 1963–1981.

[22] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. "Variational Inference: A Review for Statisticians". In: *Journal of the American Statistical Association* 112.518 (2017), pp. 859–877. DOI: 10.1080/01621459.2017.1285773.

[23] V I Bogachev, A V Kolesnikov, and K V Medvedev. "Triangular transformations of measures". In: *Sbornik: Mathematics* 196.3 (Apr. 2005), p. 309.

[24] Vladimir Igorevich Bogachev and Maria Aparecida Soares Ruas. *Measure theory*. Vol. 1. Springer, 2007.

[25] B. Boll et al. "Self-Certifying Classification by Linearized Deep Assignment". In: *PAMM: Proc. Appl. Math. Mech.* 23.1 (2023), e202200169. DOI: 10.1002/pamm.202200169.

[26] Kurt D Bollacker, Steve Lawrence, and C Lee Giles. "CiteSeer: An autonomous web agent for automatic retrieval and identification of interesting publications". In: *Proceedings of the second international conference on Autonomous agents*. 1998, pp. 116–123.

[27] Rishi Bommasani et al. *On the opportunities and risks of foundation models*. 2021. arXiv: 2108.07258.

[28] Immanuel M. Bomze. "Non-cooperative two-person games in biology: A classification". In: *International Journal of Game Theory* 15.1 (Mar. 1986), pp. 31–57. ISSN: 1432-1270.

[29] N. Bonnotte. “From Knothe’s Rearrangement to Brenier’s Optimal Transport Map”. In: *SIAM J. Math. Anal.* 45.1 (2013), pp. 64–87.

[30] Endre Boros and Peter L Hammer. “Pseudo-boolean optimization”. In: *Discrete applied mathematics* 123.1-3 (2002), pp. 155–225.

[31] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. en. Oxford University Press, Feb. 2013. ISBN: 978-0-19-953525-5.

[32] O. Bousquet et al. *From optimal transport to generative modeling: the VEGAN cookbook*. 2017. arXiv: 1705.07642.

[33] Olivier Bousquet and André Elisseeff. “Stability and generalization”. In: *The Journal of Machine Learning Research* 2 (2002), pp. 499–526.

[34] P. Brèmaud. *Discrete Probability Models and Methods*. Springer, 2017.

[35] Tom Brown et al. “Language models are few-shot learners”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 1877–1901.

[36] Théophile Cantelobre et al. *A PAC-Bayesian perspective on structured prediction with implicit loss embeddings*. 2020. arXiv: 2012.03780.

[37] G. Carlier, A. Galichon, and F. Santambrogio. “From Knothe’s Transport to Brenier’s Map and a Continuation Method for Optimal Transport”. In: *SIAM Journal on Mathematical Analysis* 41.6 (2010), pp. 2554–2576.

[38] Oliver Catoni. *PAC-Bayesian Supervised Classification: The Thermodynamics of Statistical Learning*. Vol. 56. IMS Lecture Notes Monograph Series. Institute of Mathematical Statistics, 2007.

[39] Olivier Catoni. “A PAC-Bayesian approach to adaptive classification”. In: *preprint* 840 (2003), p. 2.

[40] Marc Chamberland and Ross Cressman. “An Example of Dynamic (In)Consistency in Symmetric Extensive Form Evolutionary Games”. In: *Games and Economic Behavior* 30.2 (2000), pp. 319–326. ISSN: 0899-8256.

[41] R. T. Q. Chen, B. Amos, and M. Nickel. “Semi-Discrete Normalizing Flows through Differentiable Tesselation”. In: *Advances in Neural Information Processing Systems*. 2022.

[42] R. T. Q. Chen and Y. Lipman. *Riemannian Flow Matching on General Geometries*. 2023. arXiv: 2302.03660.

[43] R. T. Q. Chen et al. “Neural Ordinary Differential Equations”. In: *Advances in Neural Information Processing Systems*. 2018.

[44] T. Chen, R. Zhang, and G. Hinton. “Analog Bits: Generating Discrete Data Using Diffusion Models with Self-Conditioning”. In: *International Conference on Learning Representations*. 2023.

[45] Carlo Ciliberto, Francis Bach, and Alessandro Rudi. "Localized Structured Prediction". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019.

[46] Carlo Ciliberto, Lorenzo Rosasco, and Alessandro Rudi. "A general framework for consistent structured prediction with implicit loss embeddings". In: *The Journal of Machine Learning Research* 21.1 (2020), pp. 3852–3918.

[47] Eugenio Clerico, George Deligiannidis, and Arnaud Doucet. "Conditionally Gaussian PAC-Bayes". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2022, pp. 2311–2329.

[48] Eugenio Clerico, George Deligiannidis, and Arnaud Doucet. "Wide stochastic networks: Gaussian limit and PAC-Bayesian training". In: *Proceedings of The 34th International Conference on Algorithmic Learning Theory*. Ed. by Shipra Agrawal and Francesco Orabona. Vol. 201. Proceedings of Machine Learning Research. PMLR, Feb. 2023, pp. 447–470.

[49] Marius Cordts et al. "The Cityscapes Dataset for Semantic Urban Scene Understanding". In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[50] Ross Cressman. *The stability concept of evolutionary game theory: a dynamic approach*. Vol. 94. Springer, 1992.

[51] Ross Cressman and Yi Tao. "The replicator equation and other game dynamics". In: *Proceedings of the National Academy of Sciences* 111.supplement_3 (2014), pp. 10810–10817.

[52] V. De Bortoli et al. "Riemannian Score-Based Generative Modelling". In: *Advances in Neural Information Processing Systems*. 2022.

[53] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. IEEE. 2009, pp. 248–255.

[54] Jacob Devlin et al. *Bert: Pre-training of deep bidirectional transformers for language understanding*. 2018. arXiv: 1810.04805.

[55] Luc Devroye and Terry Wagner. "Distribution-free inequalities for the deleted and holdout error estimates". In: *IEEE Transactions on Information Theory* 25.2 (1979), pp. 202–207.

[56] Prafulla Dhariwal and Alexander Quinn Nichol. "Diffusion Models Beat GANs on Image Synthesis". In: *Advances in Neural Information Processing Systems*. Ed. by A. Beygelzimer et al. 2021.

[57] J. Dick and F. Pillichshammer. *Digital Nets and Sequences: Discrepancy Theory and Quasi-Monte Carlo Integration*. Cambridge University Press, 2010.

[58] Josef Dick, Frances Y. Kuo, and Ian H. Sloan. "High-Dimensional Integration: The Quasi-Monte Carlo Way". In: *Acta Numerica* 22 (Apr. 2013), pp. 133–288.

[59] Laurent Dinh, David Krueger, and Yoshua Bengio. *NICE: Non-linear Independent Components Estimation*. Apr. 2015. arXiv: 1410.8516.

[60] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. "Density estimation using Real NVP". In: *International Conference on Learning Representations*. 2017.

[61] J. Domke. "Learning Graphical Model Parameters with Approximate Marginal Inference". In: *IEEE Trans. Pattern Analysis and Machine Intelligence* 35.10 (2013), pp. 2454–2467.

[62] Justin Domke. "Learning Graphical Model Parameters with Approximate Marginal Inference". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.10 (2013), pp. 2454–2467. DOI: 10.1109/TPAMI.2013.31.

[63] Monroe D Donsker and SRS386024 Varadhan. "Asymptotic evaluation of certain Markov process expectations for large time, II". In: *Communications on Pure and Applied Mathematics* 28.2 (1975), pp. 279–301.

[64] Mathias Drton, Bernd Sturmfels, and Seth Sullivant. *Lectures on Algebraic Statistics*. Basel: Birkhäuser Basel, 2009. ISBN: 978-3-7643-8905-5. DOI: 10.1007/978-3-7643-8905-5.

[65] E. B. Dynkin. "Sufficient Statistics and Extreme Points". In: *Ann. Probability* 6.5 (1978), pp. 705–730.

[66] Gintare Karolina Dziugaite and Daniel M Roy. "Data-dependent PAC-Bayes priors via differential privacy". In: *Advances in Neural Information Processing Systems*. Vol. 31. Advances in Neural Information Processing Systems. Curran Associates, Inc., 2018.

[67] Gintare Karolina Dziugaite and Daniel M. Roy. "Computing Nonvacuous Generalization Bounds for Deep (Stochastic) Neural Networks with Many More Parameters than Training Data". In: *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence, UAI 2017, Sydney, Australia, August 11-15, 2017*. Ed. by Gal Elidan, Kristian Kersting, and Alexander T. Ihler. AUAI Press, 2017.

[68] Doris Fiebig. "Mixing properties of a class of Bernoulli-processes". In: *Transactions of the American Mathematical Society* 338.1 (1993), pp. 479–493.

[69] H. Föllmer. "Tail structure of markov chains on infinite product spaces". In: *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete* 50.3 (Jan. 1979), pp. 273–285. ISSN: 1432-2064.

[70] Efstratios Gallopoulos and Yousef Saad. "On the parallel solution of parabolic equations". In: *Proceedings of the 3rd international conference on Supercomputing*. 1989, pp. 17–28.

[71] Roy J. Gardner. *Games for business and economics*. eng. 2. ed. Hoboken, NJ: Wiley, 2003, XX, 434 S. ISBN: 978-0-471-23071-7.

[72] Andrew Gelman et al. "A weakly informative default prior distribution for logistic and other regression models". In: *The annals of applied statistics* 2.4 (2008), pp. 1360–1383.

[73] S. Geman and D. Geman. "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images". In: *IEEE Trans. Patt. Anal. Mach. Intell.* 6.6 (1984), pp. 721–741.

[74] A. Genevay, G. Peyré, and M. Cuturi. *GAN and VAE from an Optimal Transport Point of View*. 2017. arXiv: 1706.01807.

[75] B. Gidas. "A Renormalization Group Approach to Image Processing Problems". In: *IEEE Trans. Patt. Anal. Mach. Intell.* 11.11 (1989), pp. 164–180.

[76] Daniel Gonzalez-Alvarado, Alexander Zeilmann, and Christoph Schnörr. "Quantifying Uncertainty of Image Labelings Using Assignment Flows". In: *DAGM GCPR: Pattern Recognition*. Vol. 13024. Lecture Notes in Computer Science. Springer International Publishing, Jan. 2022, pp. 453–466.

[77] Ulrich Görtz and Torsten Wedhorn. *Algebraic Geometry I: Schemes: With Examples and Exercises*. Springer Studium Mathematik - Master. Wiesbaden: Springer Fachmedien, 2020. ISBN: 978-3-658-30733-2. DOI: 10.1007/978-3-658-30733-2.

[78] Will Grathwohl et al. *Ffjord: Free-form continuous dynamics for scalable reversible generative models*. 2018. arXiv: 1810.01367.

[79] Albert Gu and Tri Dao. *Mamba: Linear-time sequence modeling with selective state spaces*. 2023. arXiv: 2312.00752.

[80] Benjamin Guedj. *A Primer on PAC-Bayesian Learning*. 2019. arXiv: 1901.05353.

[81] Benjamin Guedj and Louis Pujol. "Still No Free Lunches: The Price to Pay for Tighter PAC-Bayes Bounds". en. In: *Entropy* 23.11 (Nov. 2021), p. 1529. ISSN: 1099-4300.

[82] Maxime Haddouche et al. "PAC-Bayes unleashed: generalisation bounds with unbounded losses". In: *Entropy* 23.10 (2021), p. 1330.

[83] Ramtin Hamavar and Babak Mohammadzadeh Asl. "Seizure onset detection based on detection of changes in brain activity quantified by evolutionary game theory model". In: *Computer Methods and Programs in Biomedicine* 199 (2021), p. 105899. ISSN: 0169-2607.

[84] Peter Hammerstein and Reinhard Selten. "Game theory and evolutionary biology". en. In: *Handbook of Game Theory with Economic Applications*. Vol. 2. Elsevier, 1994, pp. 929–993. ISBN: 978-0-444-89427-4.

[85] Godfrey H Hardy. "On double Fourier series, and especially those which represent the double zeta-function with real and incommensurable parameters". In: *Quart. J. Math* 37.1 (1906), pp. 53–79.

[86] Patrick T. Harker and Jong-Shi Pang. "Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications". In: *Mathematical Programming* 48.1 (Mar. 1990), pp. 161–220. ISSN: 1436-4646.

[87] Koh Hashimoto. "Unpredictability induced by unfocused games in evolutionary game dynamics". en. In: *Journal of Theoretical Biology* 241.3 (Aug. 2006), pp. 669–675. ISSN: 00225193.

[88] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778.

[89] T. Heskes. "Convexity Arguments for Efficient Minimization of the Bethe and Kikuchi Free Energies". In: *J. Artif. Intell. Res.* 26 (2006), pp. 153–190.

[90] J. Ho et al. "Cascaded Diffusion Models for High Fidelity Image Generation". In: *J. Machine Learning Research* 23 (2022), pp. 1–33.

[91] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models". In: *Advances in neural information processing systems* 33 (2020), pp. 6840–6851.

[92] Jonathan Ho et al. "Flow++: Improving flow-based generative models with variational dequantization and architecture design". In: *International conference on machine learning*. PMLR. 2019, pp. 2722–2730.

[93] Wassily Hoeffding. "Probability Inequalities for Sums of Bounded Random Variables". In: *Journal of the American Statistical Association* 58.301 (1963), pp. 13–30. DOI: 10.1080/01621459.1963.10500830.

[94] J. Hofbauer. "From Nash and Brown to Maynard Smith: Equilibria, Dynamics and ESS". In: *Selection* 1.1-3 (2001), pp. 81–88.

[95] Josef Hofbauer and Karl Sigmund. *Evolutionary games and population dynamics*. Cambridge university press, 1998.

[96] E. Hoogeboom et al. "Argmax Flows and Multinomial Diffusion: Learning Categorial Distributions". In: *Advances in Neural Information Processing Systems*. 2021.

[97] C.-W. Huang et al. "Riemannian Diffusion Models". In: *Advances in Neural Information Processing Systems*. 2022.

[98] Ruben Hühnerbein et al. "Learning Adaptive Regularization for Image Labeling Using Geometric Assignment". In: *Journal of Mathematical Imaging and Vision* 63.2 (Aug. 2020), pp. 186–215.

[99] Michael F Hutchinson. "A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines". In: *Communications in Statistics-Simulation and Computation* 18.3 (1989), pp. 1059–1076.

[100] Ernst Ising. "Beitrag zur Theorie des Ferromagnetismus". In: *Zeitschrift für Physik* 31.1 (Feb. 1925), pp. 253–258. ISSN: 0044-3328. DOI: 10.1007/BF02980577.

[101] J. Jiao et al. "Minimax Estimation of Functionals of Discrete Distributions". In: *IEEE Transactions on Information Theory* 61.5 (2015), pp. 2835–2885.

[102] M. I. Jordan, T. S. Ghahramani Z. abd Jaakkola, and L. K. Saul. "An Introduction to Variational Methods for Graphical Models". In: *Machine Learning* 37 (1999), pp. 183–233.

[103] Jürgen Jost. *Riemannian Geometry and Geometric Analysis*. 7th. Springer, 2017.

[104] Kenji Kawaguchi, Leslie Pack Kaelbling, and Yoshua Bengio. *Generalization in deep learning*. 2017. arXiv: 1710.05468.

[105] Michael Kearns and Dana Ron. "Algorithmic stability and sanity-check bounds for leave-one-out cross-validation". In: *Proceedings of the tenth annual conference on Computational learning theory*. 1997, pp. 152–162.

[106] D. P Kingma and J. Ba. *Adam: A Method for Stochastic Optimization*. 2014. arXiv: 1412.6980.

[107] Diederik P Kingma and Max Welling. *Auto-encoding variational bayes*. 2013. arXiv: 1312.6114.

[108] Herbert Knothe. "Contributions to the theory of convex bodies." In: *Michigan Mathematical Journal* 4.1 (1957), pp. 39–52.

[109] I. Kobyzev, S.J. D. Prince, and M. A. Brubaker. "Normalizing Flows: An Introduction and Review of Current Methods". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 43.11 (2021), pp. 3964–3979.

[110] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[111] Aryeh Kontorovich and Maxim Raginsky. "Concentration of Measure Without Independence: A Unified Approach Via the Martingale Method". en. In: *Convexity and Concentration*. Ed. by Eric Carlen, Mokshay Madiman, and Elisabeth M. Werner. Vol. 161. Series Title: The IMA Volumes in Mathematics and its Applications. New York, NY: Springer New York, 2017, pp. 183–210. ISBN: 978-1-4939-7004-9.

[112] Leonid (Aryeh) Kontorovich and Kavita Ramanan. "Concentration Inequalities for Dependent Random Variables via the Martingale Method". In: *The Annals of Probability* 36.6 (2008), pp. 2126–2158. ISSN: 00911798.

[113] M Krause. "Fouriersche Reihen mit zwei veränderlichen Grössen". In: *Ber. Sächs. Akad. Wiss. Leipzig* 55 (1903), pp. 164–197.

[114] Agustinus Kristiadi, Matthias Hein, and Philipp Hennig. "Being bayesian, even just a bit, fixes overconfidence in relu networks". In: *International conference on machine learning*. PMLR. 2020, pp. 5436–5446.

[115] Alex Krizhevsky, Geoffrey Hinton, et al. "Learning multiple layers of features from tiny images". In: (2009).

[116] Lev Davidovich Landau and Evgenii Mikhailovich Lifshitz. *Statistical Physics: Volume 5*. Vol. 5. Elsevier, 2013.

[117] John Langford and Rich Caruana. "(Not) Bounding the True Error". In: *Advances in Neural Information Processing Systems*. Vol. 14. MIT Press, 2002, pp. 809–816.

[118] Yann LeCun, Corinna Cortes, and CJ Burges. "MNIST handwritten digit database". In: *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist* 2 (2010).

[119] M. Ledoux. *The Concerntration of Measure Phenomenon*. Amer. Math. Soc., 2001.

[120] John M. Lee. *Riemannian manifolds. an introduction to curvature*. eng. Vol. 2. Graduate texts in mathematics. New York ; Berlin ; Heidelberg [u.a.]: Springer-Verlag, 1997, XV, 224 Pages. ISBN: 0-387-98271-X.

[121] Olof Leimar and John M. McNamara. "Game theory in biology: 50 years and onwards". en. In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 378.1876 (May 2023), p. 20210509. ISSN: 0962-8436, 1471-2970.

[122] G Khaciyan Leonid. "A polynomial algorithm for linear programming". In: *Doklady Akademii Nauk SSSR* 244 (1979), pp. 1093–1096.

[123] Gaël Letarte et al. "Dichotomize and generalize: PAC-Bayesian binary activated deep neural networks". In: *Advances in Neural Information Processing Systems* 32 (2019).

[124] Mengtian Li, Alexander Shekhovtsov, and Daniel Huber. "Complexity of discrete energy minimization problems". In: *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer. 2016, pp. 834–852.

[125] Y. Lipman et al. "Flow Matching for Generative Modeling". In: *International Conference on Learning Representations*. 2023.

[126] Ben London, Bert Huang, and Lise Getoor. "Stability and Generalization in Structured Prediction". In: *Journal of Machine Learning Research* 17.221 (2016), pp. 1–52.

[127] Ben London et al. "Collective stability in structured prediction: Generalization from one example". In: *International Conference on Machine Learning*. PMLR. 2013, pp. 828–836.

[128] Ben London et al. "PAC-Bayesian collective stability". In: *Artificial Intelligence and Statistics*. PMLR. 2014, pp. 585–594.

[129] Ilya Loshchilov and Frank Hutter. "SGDR: Stochastic Gradient Descent with Warm Restarts". In: *International Conference on Learning Representations*. 2017.

[130] A. Lou et al. "Neural Manifold Ordinary Differential Equations". In: *Advances in Neural Information Processing Systems*. 2020.

[131] Cheng Lu et al. "Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 5775–5787.

[132] Ulrike Von Luxburg and Bernhard Schölkopf. "Statistical Learning Theory: Models, Concepts, and Results". en. In: *Handbook of the History of Logic*. Vol. 10. Elsevier, 2011, pp. 651–706. ISBN: 978-0-444-52936-7. DOI: 10.1016/B978-0-444-52936-7.50016-1.

[133] Dario Madeo and Chiara Mocenni. "Game Interactions and Dynamics on Networked Populations". In: *IEEE Transactions on Automatic Control* 60.7 (2015), pp. 1801–1810.

[134] Dario Madeo et al. “An evolutionary game theory model of spontaneous brain functioning”. In: *Scientific reports* 7.1 (2017), pp. 1–11.

[135] Dario Madeo et al. “Evolutionary game for task mapping in resource constrained heterogeneous environments”. In: *Future Generation Computer Systems* 108 (2020), pp. 762–776. ISSN: 0167-739X.

[136] Youssef Marzouk et al. “Sampling via Measure Transport: An Introduction”. In: *Handbook of Uncertainty Quantification*. Ed. by Roger Ghanem, David Higdon, and Houman Owhadi. Cham: Springer International Publishing, 2016, pp. 1–41. ISBN: 978-3-319-11259-6.

[137] E. Mathieu and M. Nickel. “Riemannian Continuous Normalizing Flows”. In: *Advances in Neural Information Processing Systems*. 2020.

[138] Andreas Maurer. *A note on the PAC Bayesian theorem*. 2004. arXiv: 0411099.

[139] Sokhna Diarra Mbacke, Florence Clerc, and Pascal Germain. “PAC-Bayesian generalization bounds for adversarial generative models”. In: *Proceedings of the 40th International Conference on Machine Learning*. ICML’23. JMLR.org, 2023.

[140] Sokhna Diarra Mbacke, Florence Clerc, and Pascal Germain. “Statistical guarantees for variational autoencoders using PAC-Bayesian theory”. In: *Advances in Neural Information Processing Systems* 36 (2024).

[141] Sokhna Diarra Mbacke and Omar Rivasplata. *A Note on the Convergence of Denoising Diffusion Probabilistic Models*. en. Dec. 2023. arXiv: 2312.05989.

[142] David A McAllester. “PAC-Bayesian model averaging”. In: *Proceedings of the twelfth annual conference on Computational learning theory*. 1999, pp. 164–170.

[143] David A. McAllester. “Some PAC-Bayesian Theorems”. In: *Machine Learning* 37.3 (Dec. 1999), pp. 355–363. ISSN: 1573-0565.

[144] M. Mézard and A. Montanari. *Information, Physics, and Computation*. Oxford Univ. Press, 2009.

[145] G. Miller. “Note on the Bias of Information Estimates”. In: *Information Theory in Psychology: Problems and Methods* (1955).

[146] Tom Minka et al. *Divergence measures and message passing*. Tech. rep. Technical report, Microsoft Research, 2005.

[147] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

[148] Stephen Montgomery-Smith and Thomas Schürmann. *Unbiased estimators for entropy and class number*. 2014. arXiv: 1410.5002.

[149] Vaishnavh Nagarajan and J Zico Kolter. “Uniform convergence may be unable to explain generalization in deep learning”. In: *Advances in Neural Information Processing Systems* 32 (2019).

[150] Anna Nagurney. *Network economics: A variational inequality approach*. Vol. 10. Springer Science & Business Media, 1998.

[151] Preetum Nakkiran et al. "Deep double descent: Where bigger models and more data hurt". In: *Journal of Statistical Mechanics: Theory and Experiment* 2021.12 (2021), p. 124003.

[152] M. A. Nowak. *Evolutionary Dynamics: Exploring the Equations of Life*. Harvard Univ. Press, 2006.

[153] Stanley Osher and Ronald Fedkiw. *Level set methods and dynamic implicit surfaces*. Springer, 2003.

[154] Art B Owen. "Multidimensional variation for quasi-Monte Carlo". In: *Contemporary Multivariate Analysis And Design Of Experiments: In Celebration of Professor Kai-Tai Fang's 65th Birthday*. World Scientific, 2005, pp. 49–74.

[155] P. Pakzad and V. Anantharam. "Estimation and Marginalization using Kikuchi Approximation Methods". In: *Neural Computation* 17.8 (2005), pp. 1836–1873.

[156] L. Paninski. "Estimation of Entropy and Mutual Information". In: *Neural computation* 15.6 (2003), pp. 1191–1253.

[157] G. Papamakarios et al. "Normalizing Flows for Probabilistic Modeling and Inference". In: *J. Machine Learning Research* 22.57 (2021), pp. 1–64.

[158] A. Paszke et al. "Pytorch: An Imperative Style, High-performance Deep Learning Library". In: *Advances in Neural Information Processing Systems* (2019).

[159] R. K. Pathria and P. D. Beale. *Statistical Mechanics*. 3rd. Academic Press, 2011.

[160] Maria Perez-Ortiz et al. *Progress in self-certified neural networks*. 2021. arXiv: 2111.07737.

[161] María Pérez-Ortiz et al. *Learning PAC-Bayes priors for probabilistic neural networks*. 2021. arXiv: 2109.10304.

[162] Marıa Pérez-Ortiz et al. "Tighter Risk Certificates for Neural Networks". In: *Journal of Machine Learning Research* 22.227 (2021), pp. 1–40.

[163] Massimo Poesio, Roland Stuckardt, and Yannick Versley. *Anaphora resolution*. Springer, 2016.

[164] R. B. Potts. "Some generalized order-disorder transformations". In: *Mathematical Proceedings of the Cambridge Philosophical Society* 48.1 (1952), pp. 106–109. DOI: 10.1017/S0305004100027419.

[165] Vidya Raju and PS Krishnaprasad. "A Variational Problem on the Probability Simplex". In: *2018 IEEE Conf. on Decision and Control (CDC)*. IEEE. 2018, pp. 3522–3528.

[166] A. Ramesh et al. *Hierarchical Text-Conditional Image Generation with CLIP Latents*. 2022.

[167] Danilo Rezende and Shakir Mohamed. "Variational inference with normalizing flows". In: *International conference on machine learning*. PMLR. 2015, pp. 1530–1538.

[168] Omar Rivasplata, Vikram M Tankasali, and Csaba Szepesvari. *PAC-Bayes with backprop*. 2019. arXiv: 1908.07380.

[169] Omar Rivasplata et al. "PAC-Bayes analysis beyond the usual bounds". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 16833–16845.

[170] T. Rizzo, B. Wemmenhove, and H. J. Kappen. "Cavity Approximation for Graphical Models". In: *Phys. Rev. E* 76.1 (2007).

[171] Murray Rosenblatt. "Remarks on a multivariate transformation". In: *The annals of mathematical statistics* 23.3 (1952), pp. 470–472.

[172] N. Rozen et al. "Moser Flow: Divergence-based Generative Modeling on Manifolds". In: *Advances in Neural Information Processing Systems*. 2021.

[173] L. Ruthotto and E. Haber. "An Introduction to Deep Generative Modeling". In: *GAMM Mitt.* 44.2 (2021), 24 pages.

[174] Y. Saad. "Analysis of Some Krylov Subspace Approximations to the Matrix Exponential Operator". In: *SIAM Journal on Numerical Analysis* 29.1 (1992), pp. 209–228.

[175] Tim Salimans et al. "PixelCNN++: Improving the PixelCNN with Discretized Logistic Mixture Likelihood and Other Modifications". In: *International Conference on Learning Representations*. 2017.

[176] Larry Samuelson. "Game theory in economics and beyond". In: *Journal of Economic Perspectives* 30.4 (2016), pp. 107–130.

[177] William H Sandholm. *Population games and evolutionary dynamics*. MIT press, 2010.

[178] William H. Sandholm. "Potential Games with Continuous Player Sets". In: *Journal of Economic Theory* 97.1 (2001), pp. 81–108. ISSN: 0022-0531.

[179] J. M. Sanz-Serna. "Symplectic Runge–Kutta Schemes for Adjoint Equations, Automatic Differentiation, Optimal Control, and More". In: *SIAM Review* 58.1 (2016), pp. 3–33. DOI: 10.1137/151002769.

[180] Yuzuru Sato, Eizo Akiyama, and James P. Crutchfield. "Stability and diversity in collective adaptation". In: *Physica D: Nonlinear Phenomena* 210.1 (2005), pp. 21–57. ISSN: 0167-2789.

[181] Yuzuru Sato and James P. Crutchfield. "Coupled replicator equations for the dynamics of learning in multiagent systems". In: *Phys. Rev. E* 67 (1 Jan. 2003), p. 015206.

[182] F. Savarino and C. Schnörr. "Continuous-Domain Assignment Flows". In: *Europ. J. Appl. Math.* 32.3 (2021), pp. 570–597.

[183] Fabrizio Savarino, Peter Albers, and Christoph Schnörr. "On the Geometric Mechanics of Assignment Flows for Metric Data Labeling". In: *Information Geometry* (Nov. 2023). ISSN: 2511-249X. DOI: 10.1007/s41884-023-00120-1.

[184] Bogdan Savchynskyy. "Discrete Graphical Models — An Optimization Perspective". In: *Foundations and Trends® in Computer Graphics and Vision* 11.3-4 (2019), pp. 160–429. ISSN: 1572-2740. DOI: 10.1561/0600000084.

[185] C. Schnörr. "Assignment Flows". In: *Handbook of Variational Methods for Nonlinear Geometric Data*. Ed. by P. Grohs, M. Holler, and A. Weinmann. Springer, 2020, pp. 235–260.

[186] Peter Schuster and Karl Sigmund. "Replicator dynamics". In: *Journal of theoretical biology* 100.3 (1983), pp. 533–538.

[187] Jonathan Schwarz et al. "Quantum State Assignment Flows". In: *Entropy* 25.9 (2023). ISSN: 1099-4300.

[188] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

[189] John Shawe-Taylor and Robert C Williamson. "A PAC analysis of a Bayesian estimator". In: *Proceedings of the tenth annual conference on Computational learning theory*. 1997, pp. 2–9.

[190] D. Sitenko, B. Boll, and C. Schnörr. "Assignment Flow For Order-Constrained OCT Segmentation". In: *Int. J. Computer Vision* 129 (2021), pp. 3088–3118. DOI: 10.1007/s11263-021-01520-5.

[191] J Maynard Smith and George R Price. "The logic of animal conflict". In: *Nature* 246.5427 (1973), pp. 15–18.

[192] John Maynard Smith. "Evolution and the Theory of Games". In: *Did Darwin get it right? Essays on games, sex and evolution*. Springer, 1982, pp. 202–215.

[193] Jascha Sohl-Dickstein et al. "Deep unsupervised learning using nonequilibrium thermodynamics". In: *International conference on machine learning*. PMLR. 2015, pp. 2256–2265.

[194] Y. Song et al. "Score-Based Generative Modeling Through Stochastic Differential Equations". In: *International Conference on Learning Representations*. 2021.

[195] Yang Song and Stefano Ermon. "Generative modeling by estimating gradients of the data distribution". In: *Advances in neural information processing systems* 32 (2019).

[196] Hannes Stark et al. "Dirichlet Flow Matching with Applications to DNA Sequence Design". In: *ICLR 2024 Workshop on Machine Learning for Genomics Explorations*. 2024. URL: https://openreview.net/forum?id=ehYe5bz8H3.

[197] Esteban G Tabak and Cristina V Turner. "A family of nonparametric density estimation algorithms". In: *Communications on Pure and Applied Mathematics* 66.2 (2013), pp. 145–164.

[198] Esteban G Tabak and Eric Vanden-Eijnden. "Density estimation by dual ascent of the log-likelihood". In: *Communications in Mathematical Sciences* 8.1 (2010), pp. 217–233.

[199] Peter D Taylor and Leo B Jonker. "Evolutionary stable strategies and game dynamics". In: *Mathematical biosciences* 40.1-2 (1978), pp. 145–156.

[200] G. Teschl. *Ordinary Differential Equations and Dynamical Systems*. Vol. 140. Grad. Studies Math. Amer. Math. Soc., 2012.

[201] Lucas Theis, Aäron van den Oord, and Matthias Bethge. "A note on the evaluation of generative models". In: *4th International Conference on Learning Representations, (ICLR) 2016*. Ed. by Yoshua Bengio and Yann LeCun. 2016.

[202] Niklas Thiemann et al. "A Strongly Quasiconvex PAC-Bayesian Bound". In: *Int. Conf. Algorithmic Learning Theory (ALT)*. Vol. 76. PMLR. Oct. 2017, pp. 466–492.

[203] Rajko Tomović and Miomir Vukobratović. *General sensitivity theory*. eng. Modern analytic and computational methods in science and mathematics. New York: American Elsevier Pub. Co, 1972, 258 S. ISBN: 0-444-00108-5.

[204] Vera Trajkovska et al. "Graphical Model Parameter Learning by Inverse Linear Programming". In: *Scale Space and Variational Methods in Computer Vision*. Ed. by François Lauze, Yiqiu Dong, and Anders Bjorholm Dahl. Cham: Springer International Publishing, 2017, pp. 323–334. ISBN: 978-3-319-58771-4.

[205] D. Tran et al. "Discrete Flows: Invertible Generative Models of Discrete Data". In: *Advances in Neural Information Processing Systems*. 2019.

[206] Brian L. Trippe and Richard E. Turner. *Conditional Density Estimation with Bayesian Normalising Flows*. Feb. 2018. arXiv: 1802.04908.

[207] Benigno Uria, Iain Murray, and Hugo Larochelle. "RNADE: The real-valued neural autoregressive density-estimator". In: *Advances in Neural Information Processing Systems*. Ed. by C. J. Burges et al. Vol. 26. Curran Associates, Inc., 2013.

[208] G. Valiant and P. Valiant. "Estimating the Unseen: an n/log (n)-Sample Estimator for Entropy and Support Size, shown optimal via new CLTs". In: *Proc. 43th ACM Symposium on Theory of Computing*. 2011, pp. 685–694.

[209] G. Valiant and P. Valiant. "Estimating the Unseen: Improved Estimators for Entropy and other Properties". In: *Journal of the ACM* 64.6 (2017), pp. 1–41.

[210] V. N. Vapnik and A. Ya. Chervonenkis. "On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities". In: *Theory of Probability & Its Applications* 16.2 (1971), pp. 264–280. DOI: 10.1137/1116025.

[211] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[212] Vandana Revathi Venkateswaran and Chaitanya S. Gokhale. "Evolutionary dynamics of complex multiple games". en. In: *Proceedings of the Royal Society B: Biological Sciences* 286.1905 (June 2019). ISSN: 0962-8452, 1471-2954.

[213] M. J. Wainwright, T. S. Jaakola, and A. S. Willsky. "Tree-Based Reparameterization Framework for Analysis of Sum-Product and Related Algorithms". In: *IEEE Trans. Inf. Theory* 49.5 (2003), pp. 1120–1146.

[214] M.J. Wainwright and M.I. Jordan. "Graphical Models, Exponential Families, and Variational Inference". In: *Found. Trends Mach. Learn.* 1.1-2 (2008), pp. 1–305.

[215] Yihong Wu and Pengkun Yang. "Minimax Rates of Entropy Estimation on Large Alphabets via Best Polynomial Approximation". In: *IEEE Transactions on Information Theory* 62.6 (2016), pp. 3702–3720.

[216] Han Xiao, Kashif Rasul, and Roland Vollgraf. *Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms*. Aug. 28, 2017. arXiv: cs.LG/1708.07747 [cs.LG].

[217] Ge Yang et al. "Tuning large neural networks via zero-shot hyperparameter transfer". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 17084–17097.

[218] Greg Yang. *Tensor Programs II: Neural Tangent Kernel for Any Architecture*. 2020. arXiv: 2006.14548 [stat.ML].

[219] Greg Yang. *Tensor Programs III: Neural Matrix Laws*. 2021. arXiv: 2009.10685 [cs.NE].

[220] Greg Yang. "Wide feedforward or recurrent neural networks of any architecture are gaussian processes". In: *Advances in Neural Information Processing Systems* 32 (2019).

[221] Greg Yang and Edward J. Hu. "Tensor Programs IV: Feature Learning in Infinite-Width Neural Networks". In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, July 2021, pp. 11727–11737.

[222] Greg Yang et al. *Tensor Programs VI: Feature Learning in Infinite-Depth Neural Networks*. 2023. arXiv: 2310.02244 [cs.NE].

[223] L. Yang et al. "Diffusion Models: A Comprehensive Survey of Methods and Applications". In: *ACM Computing Surveys* 56.4 (2023).

[224] Sergey Zagoruyko and Nikos Komodakis. "Wide Residual Networks". In: *Proceedings of the British Machine Vision Conference (BMVC)*. BMVA Press, Sept. 2016, pp. 87.1–87.12.

[225] A. Zeilmann et al. "Geometric Numerical Integration of the Assignment Flow". In: *Inverse Problems* 36.3 (2020), 034004 (33pp).

[226] Alexander Zeilmann, Stefania Petra, and Christoph Schnörr. "Learning Linearized Assignment Flows for Image Labeling". In: *Journal of Mathematical Imaging and Vision* 65.1 (Jan. 2023), pp. 164–184. ISSN: 1573-7683. DOI: 10.1007/s10851-022-01132-9.

[227] A. Zern, A. Zeilmann, and C. Schnörr. "Assignment Flows for Data Labeling on Graphs: Convergence and Stability". In: *Information Geometry* 5.2 (Nov. 2021), pp. 355–404. DOI: 10.1007/s41884-021-00060-8.

[228] A. Zern et al. "Unsupervised Assignment Flow: Label Learning on Feature Manifolds by Spatially Regularized Geometric Assignment". In: *Journal of Mathematical Imaging and Vision* 62.6–7 (Apr. 2019), pp. 982–1006.

[229] Chiyuan Zhang et al. “Understanding deep learning (still) requires rethinking generalization”. In: *Communications of the ACM* 64.3 (2021), pp. 107–115.

[230] Chiyuan Zhang et al. *Understanding deep learning requires rethinking generalization.* 2016. arXiv: 1611.03530.

[231] M. Zisler et al. “Self-Assignment Flows for Unsupervised Data Labeling on Graphs”. In: *SIAM Journal on Imaging Sciences* 13.3 (2020), pp. 1113–1156.

A Additional Lemmata

Lemma A.1 *The mapping $T\colon \mathcal{W} \to \mathcal{T}$ defined by* (4.5a) *is injective.*

Proof. Let $W^{(1)}, W^{(2)} \in \mathcal{W}$ satisfy $T(W^{(1)}) = T(W^{(2)})$. Let $\gamma \in [c]^n$ be an arbitrary fixed multi-index. Fix an arbitrary index $i \in [n]$ and let $\alpha \in [c]^n$ match γ at all positions $k \in [n] \setminus \{i\}$. Then $T(W^{(1)}) = T(W^{(2)})$ implies both $T(W^{(1)})_\gamma = T(W^{(2)})_\gamma$ and $T(W^{(1)})_\alpha = T(W^{(2)})_\alpha$. Division thus gives

$$W^{(1)}_{i,\alpha_i} W^{(2)}_{i,\gamma_i} = W^{(1)}_{i,\gamma_i} W^{(2)}_{i,\alpha_i}. \tag{A.1}$$

Since $W^{(1)}, W^{(2)} \in \mathcal{W}$, the entries of row i sum to 1. Using this and the fact that $\alpha_i \in [c]$ is arbitrary, we find

$$W^{(2)}_{i,\gamma_i} = \sum_{j\in[c]} W^{(1)}_{i,j} W^{(2)}_{i,\gamma_i} \overset{(A.1)}{=} \sum_{j\in[c]} W^{(1)}_{i,\gamma_i} W^{(2)}_{i,j} = W^{(1)}_{i,\gamma_i}. \tag{A.2}$$

Since $\gamma_i \in [c]$ was arbitrary, this shows $W^{(1)} = W^{(2)}$. □

Lemma A.2 *For every $W \in \overline{\mathcal{W}}$ one has $\gamma \in \operatorname{supp}(T(W))$ if and only if $\gamma_i \in \operatorname{supp}(W_i)$ for all $i \in [n]$.*

Proof. We directly compute

$$\begin{aligned}
\operatorname{supp}(T(W)) &= \{\gamma \in [c]^n \colon T(W)_\gamma > 0\} && \text{(A.3)}\\
&= \{\gamma \in [c]^n \colon \prod_{i\in[n]} W_{i,\gamma_i} > 0\} && \text{(A.4)}\\
&= \{\gamma \in [c]^n \colon W_{i,\gamma_i} > 0,\ \forall i \in [n]\} && \text{(A.5)}\\
&= \{\gamma \in [c]^n \colon \gamma_i \in \operatorname{supp}(W_i),\ \forall i \in [n]\}. && \text{(A.6)}
\end{aligned}$$

□

Lemma A.3 *For any $V \in \mathbb{R}^{n\times c}$ it holds $Q\Pi_0 V = \Pi_0 Q V$ for the mappings Q, Π_0 given by* (4.5b) *and* (2.33).

Proof. For arbitrary $\gamma \in [c]^n$, we compute

$$(Q\Pi_0 V)_\gamma = \sum_{i\in[n]} (\Pi_0 V_i)_{\gamma_i} = \sum_{i\in[n]} \Big(V_{i,\gamma_i} - \langle V_i, \frac{1}{c}\mathbb{1}_c\rangle\Big) = (QV)_\gamma - \langle V, \underbrace{\frac{1}{c}\mathbb{1}_{n\times c}}_{=M\mathbb{1}_{\mathcal{S}}}\rangle \tag{A.7}$$

and thus, by Lemma 4.4,

$$Q\Pi_0 V = QV - \langle QV, \frac{1}{N}\mathbb{1}_N\rangle \mathbb{1}_N = \Pi_0 QV, \tag{A.8}$$

which was the assertion. □

Lemma A.4 *Let $\gamma \in [c]^n$. The differential of T at $W \in \mathcal{W}$ in direction $V \in T_0\mathcal{W}$ is given by*

$$dT|_W[V] = \Big(dT_\gamma|_W[V]\Big)_{\gamma\in[c]^n} = T(W) \diamond Q\Big[\frac{V}{W}\Big]. \tag{A.9}$$

Proof. Suppose $\eta\colon (-\varepsilon,\varepsilon) \to \mathcal{W}$ is a smooth curve with $\eta(0) = W$ and $\dot\eta(0) = V$, for some $\varepsilon > 0$. Let $\gamma \in [c]^n$ be arbitrary and consider the component T_γ. Then

$$dT_\gamma|_W[V] = \tfrac{d}{dt} T_\gamma(\eta(t))\Big|_{t=0} = \tfrac{d}{dt} \prod_{i\in[n]} \eta_{i,\gamma_i}(t)\Big|_{t=0} \tag{A.10a}$$

$$= \sum_{k\in[n]} \dot\eta_{k,\gamma_k}(0) \prod_{i\in[n]\setminus\{k\}} \eta_{i,\gamma_i}(0) = \sum_{k\in[n]} V_{k,\gamma_k} \prod_{i\in[n]\setminus\{k\}} W_{i,\gamma_i} \tag{A.10b}$$

$$= \sum_{k\in[n]} \frac{V_{k,\gamma_k}}{W_{k,\gamma_k}} T_\gamma(W) \overset{(4.5\text{b})}{=} T_\gamma(W) Q_\gamma\Big(\frac{V}{W}\Big). \tag{A.10c}$$

Because of $dT|_W[V] = (dT_\gamma|_W[V])_{\gamma\in[c]^n}$ the expression in (A.9) directly follows. □

Lemma A.5 *It holds* $\ker Q = \{\mathrm{Diag}(d)\mathbb{1}_{n\times c}\colon d \in \mathbb{R}^n,\ \langle d, \mathbb{1}_n\rangle = 0\}$ *as well as* $\operatorname{rank} Q = nc - (n-1)$.

Proof. Let $V \in \ker Q$ and let $\gamma, \tilde\gamma$ be two multi-indices which differ exactly at position k but are otherwise arbitrary. We have $(QV)_\gamma = (QV)_{\tilde\gamma} = 0$ because $V \in \ker Q$. Thus

$$(QV)_{\tilde\gamma} = V_{k,\tilde\gamma_k} + \sum_{i\in[n]\setminus\{k\}} V_{i,\tilde\gamma_i} = (QV)_\gamma = V_{k,\gamma_k} + \sum_{i\in[n]\setminus\{k\}} V_{i,\gamma_i} \tag{A.11}$$

which implies $V_{k,\tilde\gamma_k} = V_{k,\gamma_k}$, i.e. $V = \mathrm{Diag}(d)\mathbb{1}_{n\times c}$ for some $d \in \mathbb{R}^n$ since k was arbitrary. Further, it holds

$$0 = (QV)_\gamma = \sum_{i\in[n]} V_{i,\gamma_i} = \sum_{i\in[n]} d_i = \langle d, \mathbb{1}_n\rangle. \tag{A.12}$$

Thus, we have shown

$$\ker Q \subseteq \{\mathrm{Diag}(d)\mathbb{1}_{n\times c}\colon d \in \mathbb{R}^n,\ \langle d, \mathbb{1}_n\rangle = 0\}. \tag{A.13}$$

Conversely, let V be in the right-hand set. Then

$$(QV)_\gamma = \sum_{i\in[n]} V_{i,\gamma_i} = \sum_{i\in[n]} d_i = \langle d, \mathbb{1}_n\rangle = 0 \tag{A.14}$$

for all $\gamma \in [c]^n$ which shows that (A.13) is an equation. There are $(n-1)$ linearly independent vectors $d \in \mathbb{R}^n$ with $\langle d, \mathbb{1}_n\rangle = 0$, therefore Q has the specified rank. □

B Likelihood under Discrete Generative Models

For $\gamma \in [c]^n$, consider the tangent vector $U^{(\gamma)} \in T_0\mathcal{W}$ which corresponds to a smoothed version of the extremal point $M\delta_\gamma$ as defined in (5.49). We will construct a proposal distribution $\zeta \in \mathcal{P}(\mathcal{W})$ which has full support but is concentrated on the Voronoi cell $A_\gamma \subseteq \mathcal{W}$ with anchor point $M\delta_\gamma$. To this end, find $\widetilde{\zeta} \in \mathcal{P}(T_0\mathcal{W})$ such that $\widetilde{\zeta}$ has full support but is concentrated on $\widetilde{A}_\gamma = \exp_{\mathbb{1}_\mathcal{W}}^{-1}(A_\gamma)$ defined in (5.48). This is equivalent to the original task by setting $\zeta = (\exp_{\mathbb{1}_\mathcal{W}})_\sharp\widetilde{\zeta}$.

Let $B \in \mathbb{R}^{c\times(c-1)}$ be an orthonormal basis of the linear subspace $T_0\mathcal{S}_c \subseteq \mathbb{R}^c$ and let $\mathcal{B}$ be the linear operator which maps coordinates in $\mathbb{R}^{n\times(c-1)}$ to tangent vectors in $T_0\mathcal{W}$ by applying B node-wise.

We choose $\widetilde{\zeta} = \mathcal{B}_\sharp\mathcal{N}$ as a normal distribution in the basis $\mathcal{B}$ centered at $U^{(\gamma)}$ with variance $\sigma^2 > 0$ on each coordinate. In order to gain control over how concentrated ζ will be on A_γ, we compute the probability of $\widetilde{A}_\gamma$ under $\widetilde{\zeta}$ as a function of σ^2.

Independently for the tangent space of every individual simplex with index $i \in [n]$, the chosen proposal distribution $\widetilde{\zeta}_i \in \mathcal{P}(T_0\mathcal{S}_c)$ reads

$$\widetilde{\zeta}_i = B_\sharp\mathcal{N}\Big(B^\top(U^{(\gamma)})_i, \sigma^2\mathbb{I}_{c-1}\Big). \tag{B.1}$$

Let

$$D_i^{(\gamma)} = \{u_i \in T_0\mathcal{S}_c \colon \|u_i - (U^{(\gamma)})_i\|_2 \leq r\} \tag{B.2}$$

denote the ball with radius $r > 0$ in $T_0\mathcal{S}_c$ centered at $(U^{(\gamma)})_i$. For any $u_i \in T_0\mathcal{S}_c$, it holds $BB^\top u_i = u_i$ and we have

$$\begin{aligned}
\|u_i - (U^{(\gamma)})_i\|_2^2 &= \langle u_i - (U^{(\gamma)})_i, u_i - (U^{(\gamma)})_i\rangle && \text{(B.3a)}\\
&= \langle BB^\top(u_i - (U^{(\gamma)})_i), u_i - (U^{(\gamma)})_i\rangle && \text{(B.3b)}\\
&= \langle B^\top(u_i - (U^{(\gamma)})_i), B^\top(u_i - (U^{(\gamma)})_i)\rangle && \text{(B.3c)}\\
&= \|B^\top u_i - B^\top(U^{(\gamma)})_i)\|_2^2. && \text{(B.3d)}
\end{aligned}$$

Thus, $u_i \in D_i^{(\gamma)}$ exactly if the coordinates $B^\top u_i$ lie in the ball

$$\widehat{D}_i^{(\gamma)} = \{x \in \mathbb{R}^{c-1} \colon \|x - B^\top(U^{(\gamma)})_i\|_2 \leq r\}. \tag{B.4}$$

This allows to view the probability of $D_i^{(\gamma)}$ under $\widetilde{\zeta}_i$ as the probability of $\widehat{D}_i^{(\gamma)}$ under a normal distribution with variance σ^2 centered at $B^\top (U^{(\gamma)})_i$. By first shifting the mean, this can be computed as probability of the sphere $\{x \in \mathbb{R}^{c-1} \colon \|x\|_2 \leq r\}$. Let X be a standard normal random variable on $\mathbb{R}^{c-1}$, then the sought probability is

$$\mathbb{P}_{\widetilde{\zeta}_i}(D_i^{(\gamma)}) = \mathbb{P}(\|\sigma X\|_2^2 \leq r^2) = \mathbb{P}\Big(\|X\|_2^2 \leq \frac{r^2}{\sigma^2}\Big). \tag{B.5}$$

Since X has normal distribution, $\|X\|_2^2$ has χ^2-distribution and (B.5) can be computed by evaluating the cumulative distribution function Ψ_{c-1} of χ^2 with $c-1$ degrees of freedom.

Choose the radius r as the largest radius such that $D^{(\gamma)} \subseteq \widetilde{A}_\gamma$. A simple geometric argument shows that this radius is

$$r = \|U^{(\gamma)}\|_2 \sqrt{\frac{c}{2(c-1)}}. \tag{B.6}$$

We can now set a fixed probability $p > 0$ for the event that priority samples drawn from $\widetilde{\zeta}$ lie within $\widehat{D}^{(\gamma)}$. Since $\widetilde{\zeta}$ is composed of independent node-wise distributions $\widetilde{\zeta}_i$, we have $p = \widehat{p}^n$ for $\widehat{p} = \mathbb{P}_{\widetilde{\zeta}_i}(D_i^{(\gamma)})$. From (B.5), we find σ^2 as a function of p by

$$\sigma^2 = r^2 \Big(\Psi_{c-1}^{-1}\Big(p^{\frac{1}{n}}\Big)\Big)^{-1}. \tag{B.7}$$

In our experiments, we set $p = 0.5$ for likelihood computation. The above construction is crucial for robust implementation of the proposed importance sampling scheme in high dimensions.

C Additional Aspects of Assignment Flows

C.1 Stability

Theorem C.1 (Lipschitz Bound on AF) *Let $0 < T < \infty$ be a fixed time horizon and let $\overline{v}(t; v_0)$ denote the integral curve of*

$$\dot{v}(t) = \Pi_0^{\mathfrak{v}} \Omega \exp_{\mathbb{1}_{\mathcal{W}}}^{\mathfrak{v}}(v(t)), \qquad v(0) = v_0. \tag{C.1}$$

Then the map $\Psi_T \colon T_0\mathcal{W} \to T_0\mathcal{W}$, $v_0 \mapsto \Psi_T(v_0) := \overline{v}(T; v_0)$ has Lipschitz constant L_{Ψ_T} bounded by

$$L_{\Psi_T} \leq \exp\Big(\frac{1}{2}\|\Omega\|_2 T\Big). \tag{C.2}$$

The system (C.1) *is a parameterization of* (6.8) *on the tangent space $T_0\mathcal{W}$.*

Proof. Let $s(t) := \exp_{\mathbb{1}_{\mathcal{W}}}^{\mathfrak{v}}(v(t))$ for $t \geq 0$. Then (C.1) gives

$$\dot{s}(t) = \Pi_0^{\mathfrak{v}} \mathcal{R}^{\mathfrak{v}}_{\exp_{\mathbb{1}_{\mathcal{W}}}^{\mathfrak{v}}(v(t))} \dot{v}(t) = \mathcal{R}^{\mathfrak{v}}_{s(t)} \Omega s(t), \qquad s(0) = \exp_{\mathbb{1}_{\mathcal{W}}}^{\mathfrak{v}}(v(0)) \tag{C.3}$$

so (C.1) is indeed a parameterization of (6.8) in $T_0\mathcal{W}$. We formally apply the explicit Euler scheme to discretize (C.1). Let

$$f(v) := v + h\Pi_0^{\mathfrak{v}} \Omega \exp_{\mathbb{1}_{\mathcal{W}}}^{\mathfrak{v}}(v) \tag{C.4}$$

denote a single explicit Euler step. Now choose $h > 0$ such that $N = \frac{T}{h}$ is an integer and let

$$\Phi_{T,h} := f \circ \ldots \circ f = f^N \tag{C.5}$$

denote the approximation of $\overline{v}(T; v_0)$ computed by the explicit Euler scheme. We first bound $\|R_p\|_2$ for arbitrary $p \in \mathcal{W}$ by using Gerschgorins circle theorem. Each row i of R_p defines the Gerschgorin disc

$$B_{r_i}(x_i), \quad x_i = p_i(1 - p_i), \quad r_i = \sum_{j \neq i} p_i p_j = p_i(1 - p_i) \tag{C.6}$$

Since $p_i(1-p_i) \in (0, 1/4]$, the union of all discs is contained in $(-1/4, 1/2]$. This gives the upper bound

$$\|R_p\|_2 \leq \frac{1}{2}. \tag{C.7}$$

By using this together with $\|\Pi_0\|_2 = 1$, we find

$$\begin{aligned} \|\mathrm{d}f(v)\|_2 &= \|\mathbb{I} + h\Pi_0^{\mathfrak{v}}\Omega\mathcal{R}^{\mathfrak{v}}_{\exp^{\mathfrak{v}}_{\mathbb{1}_{\mathcal{W}}}(v)}\|_2 && \text{(C.8a)} \\ &\leq 1 + h\|\Pi_0^{\mathfrak{v}}\Omega\mathcal{R}^{\mathfrak{v}}_{\exp^{\mathfrak{v}}_{\mathbb{1}_{\mathcal{W}}}(v)}\|_2 && \text{(C.8b)} \\ &\leq 1 + \frac{h}{2}\|\Omega\|_2 && \text{(C.8c)} \end{aligned}$$

which in turn yields

$$\|\mathrm{d}\Phi_{T,h}(v)\|_2 \leq \|\mathrm{d}f(v)\|_2^N \leq \Big(1 + \frac{h}{2}\|\Omega\|_2\Big)^N. \tag{C.9}$$

For arbitrary $c, h \in \mathbb{R}_+$, it holds

$$1 + ch \leq \exp(ch) \quad \Rightarrow \quad (1+ch)^{\frac{1}{h}} \leq \exp(c) \tag{C.10}$$

thus

$$\|\mathrm{d}\Phi_{T,h}(v)\|_2 \leq \exp\Big(\frac{1}{2}\|\Omega\|_2 T\Big). \tag{C.11}$$

By the multivariate mean value inequality, the latter is an upper bound on the Lipschitz constant of $\Phi_{T,h}$ for all h. Let $h_k := T2^{-k}$ and define the sequence of functions $(\Phi_{T,h_k})_{k\in\mathbb{N}}$. All functions in this sequence have Lipschitz constant bounded by (C.11). It remains to show that this sequence converges uniformly to Ψ_T.

Because the r.h.s. of (C.1) is smooth, the local truncation error of Euler's method with stepsize h is bounded by Ch^2 [2, Thm. II.3.1]. Analogous to (C.8) we further compute the update Lipschitz constant

$$\|\Pi_0^{\mathfrak{v}}\Omega\mathcal{R}^{\mathfrak{v}}_{\exp^{\mathfrak{v}}_{\mathbb{1}_{\mathcal{W}}}(v)}\|_2 \leq \frac{1}{2}\|\Omega\|_2 \tag{C.12}$$

which by [2, Theorem II.3.6] bounds the global truncation error of Euler's method as

$$\|\Phi_{T,h}(v_0) - \Psi_T(v_0)\|_2 \leq h\frac{2C}{\|\Omega\|_2}\Big(\exp\Big(\frac{1}{2}\|\Omega\|_2 T\Big) - 1\Big) \in \mathcal{O}(h) \tag{C.13}$$

This shows that the sequence $(\Phi_{T,h_k})_{k\in\mathbb{N}}$ converges uniformly to Ψ_T because the pointwise convergence rate $\mathcal{O}(h)$ given in (C.13) is independent of v_0. By uniform convergence, the limit function Ψ_T also has Lipschitz constant bounded by (C.11) which completes the proof. □

C.2 Numerical Integration

In its most generic form, the assignment flow reads

$$\dot{W}(t) = \mathcal{R}_{W(t)}[F(W(t))], \qquad W(0) = W_0 \tag{C.14}$$

and fixing an arbitrary point $W_1 \in \mathcal{W}$, we find the parameterization

$$W(t) = \exp_{W_1}(V(t)), \tag{C.15a}$$
$$\dot{V}(t) = \Pi_0 F(\exp_{W_1}(V(t))), \qquad V(0) = \exp_{W_1}^{-1}(W_0) \tag{C.15b}$$

in the tangent space at W_1 which is equivalent to (C.14) as can be seen by differentiating $W(t)$ in (C.15a)

$$\dot{W}(t) = \mathcal{R}_{\exp_{W_1}(V(t))}[\dot{V}(t)] = \mathcal{R}_{W(t)}[\dot{V}(t)] \tag{C.16a}$$
$$= \mathcal{R}_{W(t)}[\Pi_0 F(\exp_{W_1}(V(t)))] \tag{C.16b}$$
$$= \mathcal{R}_{W(t)}[F(W(t))] \tag{C.16c}$$

and comparing initializations

$$W(0) = \exp_{W_1}(V(0)) = \exp_{W_1}(\exp_{W_1}^{-1}(W_0)) = W_0. \tag{C.17}$$

Due to the lifting map property, we have

$$\exp_W(V) = \exp_{\exp_{W_1}(\exp_{W_1}^{-1}(W))}(V) = \exp_{W_1}(V + \exp_{W_1}^{-1}(W)). \tag{C.18}$$

On a single simplex, consider the barycenter $p_0 = \mathbb{1}_{\mathcal{S}_c}$. Then

$$\exp_{p_0}(v) = \frac{p_0 \diamond e^v}{\langle p_0, e^v \rangle} = \frac{\frac{1}{c}\mathbb{1}_c \diamond e^v}{\langle \frac{1}{c}\mathbb{1}_c, e^v \rangle} = \frac{\mathbb{1}_c \diamond e^v}{\langle \mathbb{1}_c, e^v \rangle} = \operatorname{softmax}(v) \tag{C.19a}$$
$$\exp_{p_0}^{-1}(p) = \Pi_0 \log p \tag{C.19b}$$

The tangent space parameterization at W_0 reads

$$W(t) = \exp_{W_0}(V(t)), \qquad V(0) = 0 \tag{C.20a}$$
$$\dot{V}(t) = \Pi_0 F(\exp_{W_0}(V(t))) \tag{C.20b}$$

If, by comparison, we parameterize the same flow in the tangent space at the barycenter, this reads

$$W(t) = \exp_{\mathbb{1}_\mathcal{W}}(V^0(t)), \qquad V^0(0) = \exp_{\mathbb{1}_\mathcal{W}}^{-1}(W_0) \tag{C.21a}$$
$$\dot{V}^0(t) = \Pi_0 F(\exp_{\mathbb{1}_\mathcal{W}}(V^0(t))) \tag{C.21b}$$

We compare (C.20) with (C.21) and find

$$\exp_{\mathbb{1}_\mathcal{W}}(V^0(t)) = W(t) = \exp_{W_0}(V(t)) = \exp_{\mathbb{1}_\mathcal{W}}(V(t) + \Pi_0 \log W_0) \tag{C.22}$$

by using (C.18) and (C.19b). Thus, at any time t, both parameterzations are related by

$$V^0(t) = V(t) + \Pi_0 \log W_0 \tag{C.23}$$

which is exactly the constant shift compensated by the initialization of $V^0(0)$. Thus, (C.20) and (C.21) are equivalent foundations of numerical methods for integrating assignment flows, in the sense that both evaluate numerically to the same vector field at any time t.

C.3 Arnoldi's Method

Theorem C.2 (Arnoldi by QR-factorization) *Let* $\mathsf{V} = \mathsf{QR}$, $\mathsf{Q} \in \mathbb{R}^{nc \times k}$, $\mathsf{R} \in \mathbb{R}^{k \times k}$ *be a QR factorization of the matrix with columns*

$$\mathsf{V} = \begin{bmatrix} v_D & \mathsf{M}v_D & \cdots & \mathsf{M}^{k-1}v_D \end{bmatrix}. \tag{C.24}$$

Then Q *and* $\mathsf{H}_k = \mathsf{Q}^\top \mathsf{M} \mathsf{Q}$ *satisfy the Arnoldi relations* (6.16) *for a vector* r_k *with* $\mathsf{Q}^\top r_k = 0$ *and* H_k *is an upper Hessenberg matrix.*

Proof. We first define the quantities

$$\overline{q} = \mathsf{M}^k v_D - \mathsf{Q}\mathsf{Q}^\top \mathsf{M}^k v_D, \qquad \overline{r} = \mathsf{Q}^\top \mathsf{M}^k v_D \tag{C.25a}$$

$$r_{k+1,k+1} = \|\overline{q}\|_2, \qquad q = \overline{q}/\|\overline{q}\|_2 \tag{C.25b}$$

and note that $\overline{q}$ is orthogonal to $\operatorname{img} Q$, because $\mathsf{Q}\mathsf{Q}^\top \mathsf{M}^k v_D$ is the orthogonal projection of $\mathsf{M}^k v_D$ to $\operatorname{img} \mathsf{Q}$. A simple computation shows that

$$\begin{bmatrix} \mathsf{V} & \mathsf{M}^k v_D \end{bmatrix} = \begin{bmatrix} \mathsf{Q} & q \end{bmatrix} \begin{bmatrix} \mathsf{R} & \overline{r} \\ 0 & r_{k+1,k+1} \end{bmatrix} = \begin{bmatrix} \mathsf{QR} & \mathsf{Q}\overline{r} + r_{k+1,k+1} q \end{bmatrix} \tag{C.26}$$

is a QR decomposition of $\begin{bmatrix} \mathsf{V} & \mathsf{M}^k v_D \end{bmatrix}$. Then

$$\mathsf{MQ} = \mathsf{MQRR}^{-1} = \begin{bmatrix} \mathsf{M}v_D & \cdots & \mathsf{M}^k v_D \end{bmatrix} \mathsf{R}^{-1} = \begin{bmatrix} \mathsf{V} & \mathsf{M}^k v_D \end{bmatrix} \begin{bmatrix} 0 \\ \mathsf{R}^{-1} \end{bmatrix} \tag{C.27a}$$

$$\overset{(C.26)}{=} \begin{bmatrix} \mathsf{QR} & \mathsf{Q}\overline{r} + r_{k+1,k+1} q \end{bmatrix} \begin{bmatrix} 0 \\ \mathsf{R}^{-1} \end{bmatrix} \tag{C.27b}$$

$$= \left(\begin{bmatrix} \mathsf{QR} & \mathsf{Q}\overline{r} \end{bmatrix} + \begin{bmatrix} 0 & r_{k+1,k+1} q \end{bmatrix} \right) \begin{bmatrix} 0 \\ \mathsf{R}^{-1} \end{bmatrix} \tag{C.27c}$$

$$= \begin{bmatrix} \mathsf{QR} & \mathsf{Q}\overline{r} \end{bmatrix} \begin{bmatrix} 0 \\ \mathsf{R}^{-1} \end{bmatrix} + \begin{bmatrix} 0 & r_{k+1,k+1} q \end{bmatrix} \begin{bmatrix} 0 \\ \mathsf{R}^{-1} \end{bmatrix} \tag{C.27d}$$

and the right summand has entries

$$\left(\begin{bmatrix} 0 & r_{k+1,k+1} q \end{bmatrix} \begin{bmatrix} 0 \\ \mathsf{R}^{-1} \end{bmatrix} \right)_{ij} = \sum_{l \in [k+1]} \begin{bmatrix} 0 & r_{k+1,k+1} q \end{bmatrix}_{il} \begin{bmatrix} 0 \\ \mathsf{R}^{-1} \end{bmatrix}_{lj} = r_{k+1,k+1} q_i \mathsf{R}^{-1}_{kj} \delta_{kj} \tag{C.28}$$

because R^{-1} is an upper triangular matrix. Substituting (C.28) into (C.27) gives

$$A\mathsf{QQ} = \begin{bmatrix} \mathsf{QR} & \mathsf{Q}\overline{r} \end{bmatrix} \begin{bmatrix} 0 \\ \mathsf{R}^{-1} \end{bmatrix} + r_{k+1,k+1} \mathsf{R}^{-1}_{kk} q e_k^\top. \tag{C.29}$$

Using the definition of H_k, we find

$$\mathsf{QH}_k = \mathsf{QQ}^\top A \mathsf{Q} = \mathsf{QQ}^\top \begin{bmatrix} \mathsf{QR} & \mathsf{Q}\overline{r} \end{bmatrix} \begin{bmatrix} 0 \\ \mathsf{R}^{-1} \end{bmatrix} + r_{k+1,k+1} \mathsf{R}^{-1}_{kk} \mathsf{Q} \underbrace{\mathsf{Q}^\top q}_{=0} e_k^\top \tag{C.30}$$

which shows

$$A\mathsf{Q} = \mathsf{Q}\mathsf{H}_k + r_{k+1,k+1}\mathsf{R}_{kk}^{-1} q e_k^\top . \tag{C.31}$$

Thus, Q and H_k satisfy the Arnoldi relations (6.16) with

$$r_k = r_{k+1,k+1}\mathsf{R}_{kk}^{-1} q \tag{C.32}$$

which is orthogonal to $\operatorname{img}\mathsf{Q}$. It remains to show that H_k is an upper Hessenberg matrix. Let $i, j \in [k]$, $i > j + 1$, then

$$(\mathsf{H}_k)_{ij} \overset{(C.30)}{=} \sum_{l \in [k+1]} \begin{bmatrix} \mathsf{R} & \overline{r} \end{bmatrix}_{il} \begin{bmatrix} 0 \\ \mathsf{R}^{-1} \end{bmatrix}_{lj} \tag{C.33}$$

and because both R and R^{-1} are upper triangular matrices, the first factor of each summand can only be nonzero for $i \leq l$ and the second factor can only be nonzero if $l \leq j + 1$. Thus, a summand can only be nonzero if $i \leq l \leq j + 1$, showing that H_k is an upper Hessenberg matrix. □

D Additional Detail of Experiments

D.1 Implementation Details

For the Cityscapes experiment in Section 5.3.5, we employ the UNet architecture of [1] with *attention_resolutions* (32, 16, 8), *channel_mult* (1,1,2,3,4), 4 attention heads, 3 blocks and 64 channels. We trained for 250 epochs using Adam with learning rate 0.0001 and cosine annealing scheduler. Rather than the original $c = 33$ classes, we only use the $c = 8$ class categories specified in *torchvision*. The same subsampling of classes was used in the related work [3]. They additionally perform spatial subsampling to 32×64. Instead, we subsample the spatial dimensions (*NEAREST* interpolation) to 128×256.

For the MNIST experiment in Section 5.3.5, we use the same architecture with *attention_resolutions* (16), *channel_mult* (1,2,2,2), 4 attention heads, 2 blocks and 32 channels. We trained for 100 epochs using Adam with learning rate 0.0005 and cosine annealing scheduler. We pad the original 28×28 images with zeros to size 32×32 to be compatible with spatial downsampling employed by the UNet architecture.

For the simple distributions in Figure 5.5, we employ a neural network composed of batch normalization, dense layers and ReLU activation. The sequence of hidden dimensions for the mixture of Gaussian and Pinwheel distributions is (256, 256). For the coupled binary variables, we use a linear function F_θ, with no batch normalization or bias. We trained for 2k steps with batch size 512 using Adam with learning rate 0.0005.

In all experiments, the smoothing constant ϵ of (5.49) is set to 0.01.

All experiments in Sections 5.2.2 and 5.3.5 were run on one of two desktop graphics cards (1x NVIDIA RTX2080ti, 1x NVIDIA RTX2060super), requiring less than 200 compute hours in total.

D.2 Additional Samples

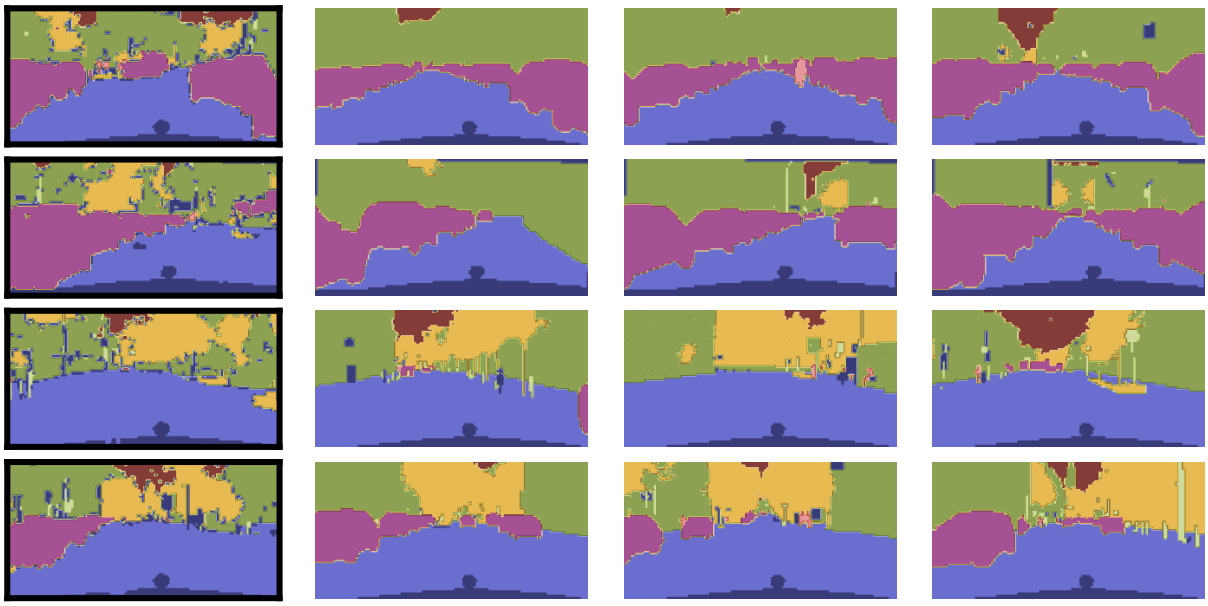

Figure D.1: Illustration of Cityscapes segmentation samples drawn from our model. *Left with a black border:* random samples generated by our model through integration (5.7) with random initializations. The remaining plots depict the five training data with smalles pixel-wise distance to the sample in their respective row.