Journal of Mathematical Imaging and Vision (2025) 67:34
https://doi.org/10.1007/s10851-025-01239-9

Generative Assignment Flows for Representing and Learning Joint Distributions of Discrete Data

Bastian Boll[1] · Daniel Gonzalez-Alvarado[1] · Stefania Petra[2] · Christoph Schnörr[1,3]

Received: 7 June 2024 / Accepted: 4 March 2025

Abstract

We introduce a novel generative model for the representation of joint probability distributions of a possibly large number of discrete random variables. The approach uses measure transport by randomized assignment flows on the statistical submanifold of factorizing distributions, which enables to represent and sample efficiently from any target distribution and to assess the likelihood of unseen data points. The complexity of the target distribution only depends on the parametrization of the affinity function of the dynamical assignment flow system. Our model can be trained in a simulation-free manner by conditional Riemannian flow matching, using the training data encoded as geodesics on the assignment manifold in a closed form, with respect to the e-connection of information geometry. Numerical experiments devoted to distributions of structured image labelings demonstrate the applicability to large-scale problems, which may include discrete distributions in other application areas. Performance measures show that our approach scales better with the increasing number of classes than the recent related work.

Keywords Generative models · Discrete distributions · Normalizing flows · Information geometry · Neural ODEs · Assignment flows · Replicator equation

Mathematics Subject Classification 49Q22 · 53B12 · 62H35 · 68T05 · 68U10 · 91A22

Contents

1 Introduction
1.1 Overview, Motivation
1.2 Related Work
1.2.1 Statistics
1.2.2 Own Prior Work
1.2.3 Machine Learning
1.3 Organization
1.4 Basic Notation, List of Main Symbols
2 Background
2.1 Assignment Flows
2.2 Meta-Simplex, Flow Embedding
3 Approach
3.1 Generative Model
3.1.1 Goal
3.1.2 Representation of General Distributions
3.1.3 Model Learning and Model Evaluation (Sampling)
3.2 Riemannian Flow Matching
3.2.1 Representation of Labelings as Training Data
3.2.2 Training Criterion
3.2.3 Constructing Conditional Fields
3.2.4 Infinite Integration Time
3.2.5 Relation to Dirichlet Flow Matching
3.3 Learning Interaction Between Simplices
3.4 Numerical Flow Integration
3.5 Likelihood Computation
3.6 Dequantization
4 Experiments and Discussion
4.1 Class Scaling
4.2 Generating Image Segmentations
4.3 Likelihood Evaluation
5 Conclusion
Appendix A. Proofs
A.1 Proofs of Section 3.2.3
A.2 Proofs of Section 3.3
Appendix B. Experiments: Details
B.1 Details of Class Scaling Experiment

✉ Daniel Gonzalez-Alvarado
daniel.gonzalez@iwr.uni-heidelberg.de
https://ipa.math.uni-heidelberg.de

[1] Institute for Mathematics, Image and Pattern Analysis Group, Heidelberg University, Heidelberg, Germany

[2] Mathematical Imaging Group, Department of Mathematics and Centre for Advanced Analytics and Predictive Sciences (CAAPS), University of Augsburg, Universitätsstr. 14, 86159 Augsburg, Germany

[3] Institute for Mathematics, Research Station Geometry and Dynamics, Heidelberg University, Heidelberg, Germany

Published online: 28 May 2025

B.2 Details of Generating Image Segmentations . . . .
B.2.1 Cityscapes Data Preparation . . . .
B.2.2 Cityscapes Training . . . .
B.2.3 Binarized MNIST Data Preparation . . . .
B.2.4 Binarized MNIST Training . . . .
Appendix C. Likelihood Computation: Details . . . .
References . . . .

1 Introduction

1.1 Overview, Motivation

Generative models in machine learning define an active area of research [32, 40, 41]. Corresponding research objectives include

(i) the representation of complex probability distributions,
(ii) efficient sampling from such distributions, and
(iii) computing the likelihoods of unseen data points.

The target probability distribution is typically not given, except for a finite sample set (empirical measure). The modeling task concerns the generation of the target distribution by transporting a simple reference measure, typically the multivariate standard normal distribution, using a corresponding pushforward mapping. This mapping is realized by a network with trainable parameters that are optimized by maximizing the likelihood of the given data or a corresponding surrogate objective which is more convenient regarding numerical optimization. This class of approaches are called *normalizing flows* in the literature.

Discrete joint probability distributions abound in applications, yet have received less attention in the literature on generative models. The recent survey paper [32] concludes with a short paragraph devoted to discrete distributions and the assessment that “the generalization of normalizing flows to discrete distributions remains an open problem”. Likewise, the survey paper [40] briefly discusses generative models of discrete distributions in [40, Section 5.3]. The authors state that “compared to flows on $\mathbb{R}^D$, discrete flows have notable theoretical limitations”. The survey paper [41] does not mention at all generative models of discrete distributions.

This paper introduces a novel generative approach for the significant subclass of *discrete* (*categorial*) probability distributions of n random variables y_i taking values in a finite set $\{1, 2, \dots, c\}$,

$$\begin{aligned} y &= (y_1, \dots, y_n)^\top \in [c]^n, \quad y_i \in [c] := \{1, 2, \dots, c\}, \\ &\quad i \in [n], \quad c, n \in \mathbb{N}. \end{aligned} \tag{1.1}$$

A corresponding distribution p is a look-up table which specifies for any realization α of the discrete random vector y the probability

$$\begin{aligned} p(\alpha) &= p(\alpha_1, \dots, \alpha_n) := \Pr(y = \alpha) \\ &= \Pr(y_1 = \alpha_1 \ \wedge \cdots \wedge \ y_n = \alpha_n), \qquad \alpha \in [c]^n. \end{aligned} \tag{1.2}$$

Any such look-up table is a nonnegative tensor with the combinatorially large number

$$N := c^n \tag{1.3}$$

of entries $p(\alpha)$, $\alpha \in [c]^n$. Furthermore, since $p(\alpha) \geq 0$, $\forall \alpha$, and $\sum_{\alpha \in [c]^n} p(\alpha) = 1$, any distribution p also corresponds to a point $p \in \Delta_N$ of the probability simplex

$$\begin{aligned} &\Delta_N := \left\{ p \in \mathbb{R}^N_{\geq 0} \colon \langle \mathbb{1}_N, p \rangle = 1 \right\}, \quad p = (p_\alpha)_{\alpha \in [c]^n}, \\ &p_\alpha := p(\alpha), \quad \text{(meta-simplex)} \end{aligned} \tag{1.4}$$

where $\mathbb{1}_N := (1, 1, \dots, 1)^\top \in \mathbb{R}^N$.

Thus, we denote with p discrete joint probability distributions using any of the equivalent representations:

- as functions $p \colon [c]^n \to [0, 1]$, cf. Eq. (1.2);
- as nonnegative tensors with c^n components $p(\alpha_1, \dots, \alpha_n)$;
- as discrete probability vectors $p \in \Delta_N$ with $N = c^n$ components p_α, where each component specifies the probability $p_\alpha = p(\alpha) = \Pr(y = \alpha)$, cf. Eq. (1.4).

In particular, the N vertices (extreme points)

$$e_\alpha \in \{0, 1\}^N \tag{1.5}$$

of Δ_N are the unit vectors, which encode the discrete Dirac measures δ_α concentrated on the realizations $\alpha \in [c]^n$.

Figure 1 illustrates the approach for the toy distribution of two binary variables, i.e., $c = 2$ and $N = 2^2 = 4$,

$$p(\alpha_1, \alpha_2) \colon \quad \begin{array}{|c||c|c|} \hline \alpha_1/\alpha_2 & 0 & 1 \\ \hline 0 & 0.45 & 0.05 \\ \hline 1 & 0.05 & 0.45 \\ \hline \end{array} \tag{1.6}$$

The simplex $\Delta_4 \subset \mathbb{R}^4$ (1.4) is visualized in $\mathbb{R}^3$ in local coordinates as tetrahedron (Fig. 1a; see Example 2.1 (p. 7) for details). The generative model only uses the submanifold of *factorizing* discrete distributions which ensures computational efficiency of both training and sampling. Figure 1(a) shows that this submanifold connects all extreme points of Δ_4.

Figure 1(b) illustrates how sampling from p is accomplished after training, by computing on the submanifold the integral curves of a generating flow which emanates from initial random points, such that each curve converges to a vertex

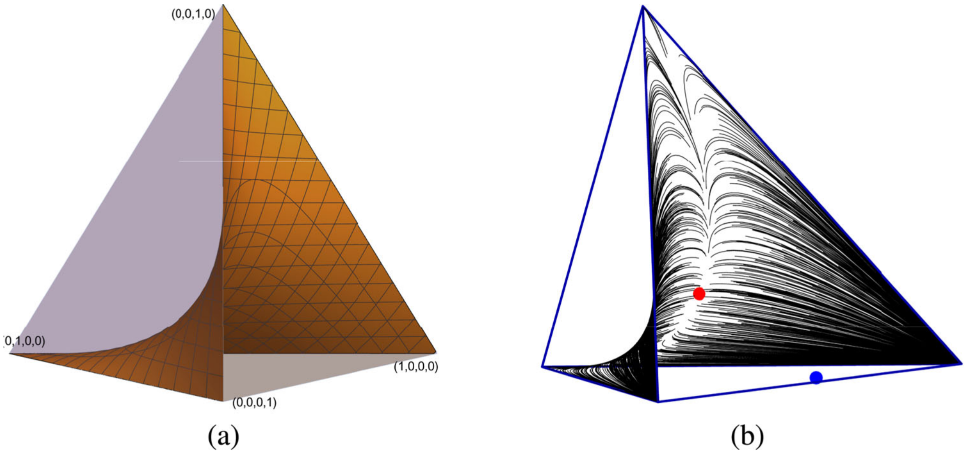


Fig. 1 **a** The simplex Δ_N (1.4), for $N = 4$, depicted in local coordinates, and the submanifold of factorizing discrete distributions which connects all extreme points of Δ_4. **b** Visualization of 1000 samples from the target distribution $p(\alpha_1, \alpha_2)$ given by (1.6), corresponding to the blue point $p \in \Delta_4$. Each sample corresponds to an integral curve of a flow, which evolves on the submanifold, and can be computed efficiently by geometric integration. The parametrized vector field of the dynamical system that generates the flow has been trained by matching the flow to geodesics on the submanifold which encode given training data. As a result, each component p_α of the target distribution corresponds to the relative frequency of integral curves converging to the vertex e_α, such that the entire distribution p is represented by the convex combination $\sum_\alpha p_\alpha e_\alpha = p$. In this way, the flow realizes the pushforward of a simple reference distribution, centered at 0 in the tangent space at the barycenter (red point), to the discrete target distribution p. Figure 2 provides a more detailed illustration of the approach

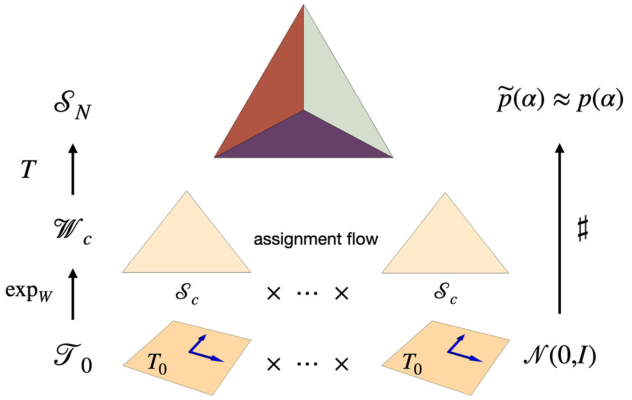


Fig. 2 Overview of the approach: The standard Gaussian reference measure $\mathcal{N}(0, I)$ is pushed forward by the lifting map $\exp_W$ from the flat tangent product space $\mathcal{T}_0$ to the assignment manifold $\mathcal{W}_c$ and further to the meta-simplex $\mathcal{S}_N$ via the embedding map T (2.7), by geometrically integrating the assignment flow equation (2.3). Since the assignment flow converges to the extreme points of $\overline{\mathcal{W}_c}$ which after embedding agree with the extreme points of $\Delta_N = \overline{\mathcal{S}_N}$, an approximation $\widetilde{p}(\alpha)$ of a general *discrete* target measure $p(\alpha)$ can be learned in terms of a corresponding convex combination of extreme points. This is achieved by matching the flow of e-geodesics that encode given training samples to the generating assignment flow, by empirical expectation, and by learning the parameters of the affinity function F_θ (1.12). Since factorizing distributions $T(W)$, $W \in \mathcal{W}_c$ are only required, the approach is computationally feasible also in high dimensions

of the simplex which represents a realization $\alpha \sim p(\alpha)$ by (1.5). In this way, a simple reference distribution is pushed forward to p. Figure 2 gives a more detailed account of the ingredients of our approach.

Training concerns the parameters of the vector field of the dynamical system, which generates the aforementioned flow on the submanifold. This is achieved by matching the flow to closed-form geodesics on the submanifold which encode given training data. This *flow-matching approach* has been recently proposed by [11, 36]. Our paper elaborates this approach for *discrete* joint probability distributions using the geometric approach outlined above.

1.2 Related Work

The central theme of our paper is large joint distributions of discrete random variables, which has been a core topic in *multivariate* and *algebraic statistics*, with numerous applications in terms of discrete graphical models in various fields. In addition, our paper contributes to research on *generative models* in *machine learning*. Related work is accordingly reported in Sects. 1.2.1 and 1.2.3, respectively, in view of own prior work briefly reported in Sect. 1.2.2, which combines both viewpoints. The recent related work discussed in Sect. 1.2.3 reflects the fact that generative models for discrete probability distributions have become an active field of research recently.

1.2.1 Statistics

Joint distributions of discrete random variables have a long history in multivariate statistics [1]. This includes the study of subsets of such distributions known as *discrete graphical* *models* [10, 31, 34]. Here, conditional independency assumptions encoded by the structure of an underlying graph [44] effectively reduce the degree of freedoms (1.3) of general discrete distributions p and imply their factorization of once realizations of conditioning variables are observed. From the algebraic viewpoint, such statistical assumptions about p give rise to monomial constraints. The study of the topology and geometry of the resulting algebraic varieties, which support corresponding subfamilies of distributions, is the subject of the fields of *algebraic statistics* [19, 22, 38, 45, 49]. The special case of fully factorizing discrete distributions

$$p(\alpha) = \prod_{i \in [n]} p_i(\alpha_i) \tag{1.7}$$

is particularly relevant for this paper. For example, the subfamily of all such distributions for the toy case $n = c = 2$, depicted by Fig. 1, is known as Wright manifold in mathematical game theory [28] and more generally as *Segre variety* $\Sigma_{1,1}$ in algebraic geometry [23, 33].

1.2.2 Own Prior Work

Our approach utilizes *assignment flows* [4] that evolve on the relative interior of the product of n probability simplices Δ_c, called *assignment manifold*, one factor for each random variable y_i, $i \in [n]$ conforming to the factorization (1.7). As summarized in Sect. 2.1, the restriction to strictly positive discrete distribution with full support enables to turn these domains into elementary statistical manifolds equipped with the Fisher–Rao geometry and the e-connection of information geometry [3]. The corresponding exponential map and the geodesics can be specified in a closed form.

Assignment flows are turned into a generative model for discrete random variables as illustrated by Fig. 2, which generalizes the toy example (1.6) and Fig. 1: Geometric integration of the assignment flow realizes a map, which pushes forward a standard reference measure on the tangent space at the barycenter to the extreme points of the (closure) of the assignment manifold. By embedding the assignment manifold into the simplex (1.4) of all discrete joint distributions, the pushforward measure concentrates on the extreme points and hence represents a more complex *non-factorizing* discrete joint distribution by convex combination of Dirac measures.

Our recent work [5] characterizes assignment flows as multi-population games and studies multi-game dynamics via the aforementioned embedding approach. Some results established in this work regarding the embedding map will be employed in Sect. 3.3.

1.2.3 Machine Learning

The lack of work on generative models for *discrete* distributions stated in the survey papers [32, 40] has stimulated the corresponding research recently.

The paper [42] employs the parametric Dirichlet distribution on the probability simplex [2, 20, 30] as intermediate conditional distributions in a flow-matching approach. A similarity to our method is the use of infinite transport time, which achieves favorable scaling in the regime of many classes. A detailed comparison is discussed in Sect. 3.2.5.

The paper [15] refers to [4] and a preliminary version [6] of our generative model and uses geodesics with respect to the Riemannian connection rather than the e-connection, corresponding to $\alpha = 0$ and $\alpha = 1$ in the family of α-connections, respectively [3]. By virtue of the sphere map [4, Def. 1] as isometry, the former geodesics on the simplex correspond to the geodesics (great circles) on the sphere with radius 2, restricted to the intersection with the open positive orthant. The authors of [15] argue that their approach avoids numerical instability at the boundary of the manifold, which is indeed relevant when working on the sphere. However, this issue does not arise on the simplex either, provided that proper geometric numerical integration schemes are used, as demonstrated in [48]. The focus of [15] is on improving the training dynamics using optimal transport, due to the close relation on the simplex of the geometry induced by the Wasserstein distance and the Fisher–Rao geometry [37].

Another line of research, called *dequantization*, concerns the approximation of *discrete* probability distributions by *continuous* distributions [18, 24, 43, 46, 47]. A dequantization approach for general discrete data, i.e., similar in scope to our approach, was recently proposed by [9]. We discuss this paper in Sect. 3.6 and point out differences by showing that our approach can be characterized as dequantization procedure. In particular, we indicate that a key component of the approach [9], learning an embedding of class configurations, can be replicated using our approach, by defining a payoff function of our generative assignment flow approach accordingly.

Regarding the training of our generative model, our approach builds on the recent work [11, 36]. The authors introduced a *flow-matching approach* to the training of continuous generative models, which enables more stable and efficient training and hence an attractive alternative to established maximum likelihood training. We adopt this criterion and adapt it to our generative model for discrete distributions and the underlying geometry. In particular, we encode the given training data as e-geodesics on the assignment manifold, which makes flow matching convenient and effective.

1.3 Organization

Section 1.4 fixes the basic notation. Section 2 summarizes the assignment flow approach and specifies the flow embedding into the simplex (1.4), along with mappings and their properties required in the remainder of this paper.

The core Sect. 3 introduces and details our approach. Section 3.1 introduces the generative model. The flow-matching approach is described in Sect. 3.2 and how it relates to the recent work [11, 36] which inspired the training component of our approach. Section 3.4 details the particular geometric integration used in all experiments for computing the assignment flow, based on the methods worked out by [48]. Section 3.5 explains how the trained generative model is evaluated for computing the likelihoods of a novel unseen data points. Section 3.6 explains dequantization and characterizes our approach from this viewpoint.

Experimental results are presented and discussed in Sect. 4. We conclude in Sect. 5.

1.4 Basic Notation, List of Main Symbols

We set $[n] := \{1, 2, \dots, n\}$ for $n \in \mathbb{N}$. The canonical Euclidean inner product and the matrix inner product, which induces the Frobenius norm, are denoted by $\langle \cdot, \cdot \rangle$. The mapping $\mathrm{Diag}(\cdot)$ takes a vector to the diagonal matrix with the vector component as main diagonal entries. e_k, $k \in \mathbb{N}$ denotes a unit vector with single nonzero k-th component equal to 1 and dimension, which is clear from the context.

Data, labelings. $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with vertex set $\mathcal{V} = [n]$, denotes an arbitrary graph on which data x_i are observed at every vertex $i \in \mathcal{V}$. $c \in \mathbb{N}$ possible class labels of the data x_i are represented by discrete random variables $y_i \in [c]$. Realizations of the variables y_i are denoted by $\alpha_i \in [c]$. This results in $N = c^n$ labeling configurations $\alpha = \{\alpha_1, \dots, \alpha_n\}$ for given data $x = \{x_1, \dots, x_n\}$.

Assignment flows, dynamical labelings. The probability simplex is denoted by:

$$\Delta_n = \{p \in \mathbb{R}^n_{\geq 0} \colon \langle \mathbb{1}_n, p \rangle = 1\}, \tag{1.8}$$

where $\mathbb{1}_n = (1, 1, \dots, 1)^\top \in \mathbb{R}^n$. *Assignment flows* (Sect. 2.1) work with the relative interior $\mathring{\Delta}_c$ of Δ_c, denoted by $\mathcal{S}_c := \mathring{\Delta}_c$, containing the strictly positive probability vectors of dimension c, and with the n-fold product conforming to $\mathcal{V}$,

$$\mathcal{W}_c := \mathcal{S}_c \times \cdots \times \mathcal{S}_c, \quad (n = |\mathcal{V}| \text{ factors}) \quad \text{(assignment manifold)} \tag{1.9}$$

Points on $\mathcal{W}_c$ are denoted by:

$$W = (W_1, \dots, W_n)^\top \in \mathcal{W}_c \subset \mathbb{R}^{n \times c}_{>0}, \quad W_i \in \mathcal{S}_c, \quad i \in [n]. \tag{1.10}$$

The evolution $W(t)$ of these assignment vectors, obtained by integrating the assignment flow equation, determines the label assignments α_i to the data point x_i at every $i \in \mathcal{V}$, by convergence to the corresponding unit vectors

$$\lim_{t \to \infty} W_i(t) = e_{\alpha_i} \in \{0, 1\}^c, \quad i \in \mathcal{V}, \tag{1.11}$$

which are the extreme points of the closure of the assignment manifold $\overline{\mathcal{W}_c}$. Further spaces and mappings defined in connection with assignment flows in Sect. 2.1 are: The tangent spaces T_0, $\mathcal{T}_0$ to $\mathcal{S}_c$, $\mathcal{W}_c$ with orthogonal projections π_0, Π_0, the barycenters $\mathbb{1}_{\mathcal{S}}$, $\mathbb{1}_{\mathcal{W}}$ of $\mathcal{S}_c$, $\mathcal{W}_c$, the Fisher–Rao metric g_p, g_W on T_0, $\mathcal{T}_0$, the replicator maps R_p, R_W and the lifting maps $\exp_p$, $\exp_W$, which play the role of exponential maps.

Besides the underlying geometry, the essential part of the assignment flow equation, whose integration results in (1.11), is the

$$F_\theta \colon \mathcal{W}_c \to \mathbb{R}^{n \times c}, \qquad \text{(affinity function)} \tag{1.12}$$

whose parameters θ are learned from data.

Meta-simplex, assignment manifold embedding. We overload the symbol p to denote discrete probability distributions using any of the equivalent representations specified after Eq. (1.4), as well as discrete probability vectors whose dimension should be unambiguous from the context. Major examples are $p \in \mathcal{S}_c \subset \mathbb{R}^c_{\geq 0}$ and $p \in \Delta_N$ (cf. (1.4)).

Since the embedding

$$\mathcal{T} := T(\mathcal{W}_c) \subset \mathcal{S}_N := \mathring{\Delta}_N \qquad \text{(meta-simplex embedding)} \tag{1.13}$$

of the assignment manifold defined in Sect. 2.2 yields the submanifold of factorizing distributions in Δ_N, as depicted for a toy scenario by Fig. 1a, we call Δ_N as defined by (1.4) "meta-simplex", to distinguish the product of simplices $\mathcal{W}_c$ (1.9) before and after the embedding $T(\mathcal{W}_c)$ (1.13).

We denote by

$$\mathcal{P}(\mathcal{S}_c), \;\; \mathcal{P}(\mathcal{W}_c), \;\; \text{etc.} \tag{1.14}$$

the set of probability measures supported on the space $\mathcal{S}_c$, $\mathcal{W}_c$, etc.

2 Background

Section 2.1 defines spaces and mappings required in the remainder of the paper. Section 2.2 defines a key ingredient of our approach, the embedding (1.13) and related mappings. We refer to the basic notation introduced in Sect. 1.4.

2.1 Assignment Flows

The basic state space of discrete distributions is the relative interior of the probability simplex:

$$\mathcal{S}_c := \mathring{\Delta}_c = \{p \in \mathbb{R}^c \colon p_j > 0,\ \langle \mathbb{1}_c, p\rangle = 1,\ \forall j \in [c]\} \tag{2.1a}$$

with its

$$\mathbb{1}_{\mathcal{S}} := \frac{1}{c}\mathbb{1}_c \in \mathcal{S}_c, \qquad \text{(barycenter)} \tag{2.1b}$$

which becomes the Riemannian manifold $(\mathcal{S}_c, g)$ with trivial tangent bundle $T\mathcal{S}_c = \mathcal{S}_c \times T_0$, comprising the

$$T_0 := T_{\mathbb{1}_{\mathcal{S}}}\mathcal{S}_c := \{v \in \mathbb{R}^c \colon \langle \mathbb{1}_c, v\rangle = 0\} \quad \text{(tangent space)} \tag{2.1c}$$

with the orthogonal projection

$$\pi_0 \colon \mathbb{R}^c \to T_0, \quad \pi_0 := I_c - \mathbb{1}_c\mathbb{1}_{\mathcal{S}}^\top \quad \text{(orthogonal projection)} \tag{2.1d}$$

and carrying the

$$g_p(u, v) := \langle u, \operatorname{Diag}(p)^{-1}v\rangle, \quad u, v \in T_0, \quad p \in \mathcal{S}_c. \qquad \text{(Fisher-Rao metric)} \tag{2.1e}$$

This naturally extends to the product manifold $(\mathcal{W}_c, g)$ given by (1.9), with trivial tangent bundle $T\mathcal{W}_c = \mathcal{W}_c \times \mathcal{T}_0$, and

$$\mathbb{1}_{\mathcal{W}} = (\mathbb{1}_{\mathcal{S}}, \dots, \mathbb{1}_{\mathcal{S}})^\top, \qquad \text{(barycenter)} \tag{2.2a}$$

$$\mathcal{T}_0 := T_{\mathbb{1}_{\mathcal{W}}}\mathcal{W}_c := T_0 \times \cdots \times T_0, \quad (n = |\mathcal{V}| \text{ factors}) \qquad \text{(tangent space)} \tag{2.2b}$$

with points denoted by

$$V = (V_1, \dots, V_n)^\top \in \mathcal{T}_0 \subset \mathbb{R}^{n\times c}, \qquad V_i \in T_0, \qquad i \in [n], \tag{2.2c}$$

the orthogonal projection

$$\Pi_0 \colon \mathbb{R}^{n\times c} \to \mathcal{T}_0, \quad \Pi_0 U := (\pi_0 U_1, \dots, \pi_0 U_n)^\top \qquad \text{(orthogonal projection)} \tag{2.2d}$$

and the

$$g_W(U, V) = \sum_{i\in[n]} g_{W_i}(U_i, V_i), \quad U, V \in \mathcal{T}_0, \quad W \in \mathcal{W}_c. \qquad \text{(Fisher-Rao metric)} \tag{2.2e}$$

Assignment flows are dynamical systems of the general form

$$\dot W(t) = R_{W(t)}\big[F_\theta\big(W(t)\big)\big], \qquad W(0) = W_0 \in \mathcal{W}_c, \qquad \text{(assignment flow)} \tag{2.3}$$

parametrized by an affinity function (1.12) and comprising the linear mappings

$$R_p \colon \mathbb{R}^c \to T_0, \quad R_p = \operatorname{Diag}(p) - pp^\top, \quad p \in \mathcal{S}_c \qquad \text{(replicator map)} \tag{2.4a}$$

$$R_W \colon \mathbb{R}^{n\times c} \to \mathcal{T}_0, \quad R_W[F_\theta] = (R_{W_1}F_{\theta,1}, \dots, R_{W_n}F_{\theta,n})^\top, \quad W \in \mathcal{W}_c. \quad \text{(replicator map)} \tag{2.4b}$$

The *exponential maps* with respect to the e-connection read

$$\operatorname{Exp}_p(v) = \frac{p\cdot e^{\frac{v}{p}}}{\langle p, e^{\frac{v}{p}}\rangle}, \qquad p \in \mathcal{S}_c, \quad v \in T_0, \tag{2.5a}$$

$$\operatorname{Exp}_W(V) = \big(\operatorname{Exp}_{W_1}(V_1), \dots, \operatorname{Exp}_{W_n}(V_n)\big)^\top \quad W \in \mathcal{W}_c, \quad V \in \mathcal{T}_0, \tag{2.5b}$$

where both the multiplication $\cdot$ and the exponential function apply componentwise. Composition with the replicator maps (2.4) yields the

$$\exp_p \colon T_0 \to \mathcal{S}_c, \quad \exp_p := \operatorname{Exp}_p \circ R_p, \quad p \in \mathcal{S}_c, \qquad \text{(lifting map)} \tag{2.6a}$$

$$\exp_W \colon \mathcal{T}_0 \to \mathcal{W}_c, \quad \exp_W := \operatorname{Exp}_W \circ R_W, \quad W \in \mathcal{W}_c. \qquad \text{(lifting map)} \tag{2.6b}$$

2.2 Meta-Simplex, Flow Embedding

The embedding (1.13) is defined by the map

$$T \colon \mathcal{W}_c \to \mathcal{T} = T(\mathcal{W}_c) \subset \mathcal{S}_N, \qquad T(W)_\alpha := \prod_{i\in[n]} W_{i,\alpha_i}, \quad \alpha \in [c]^n. \tag{2.7}$$

Denoting the tangent space to $\mathcal{S}_N$ defined by (1.13) by

$$\mathcal{T}_0\mathcal{S}_N := \{z \in \mathbb{R}^N \colon \langle \mathbb{1}_N, z\rangle = 0\}, \quad \text{(meta-tangent space)} \tag{2.8}$$

we also require the map

$$Q \colon \mathbb{R}^{n\times c} \to \mathbb{R}^N, \quad Q \colon \mathcal{T}_0 \to \mathcal{T}_0\mathcal{S}_N,$$

$$(QV)_\alpha := \sum_{i\in[n]} V_{i,\alpha_i}, \quad \alpha \in [c]^n. \tag{2.9}$$

The mappings T, Q have been studied by [5, 7].

Every point $W \in \mathcal{W}_c$ on the assignment manifold is represented through (2.7) by the combinatorially large vector $T(W)$ with $N = c^n$ components $T(W)_\alpha$, consisting of monomials of degree n in the variables $W_{i,\alpha_i} \in (0,1)$. A labeling determined by the assignment flow by (1.11) corresponds to

$$\lim_{t\to\infty} T\big(W(t)\big) = T\big((e_{\alpha_1},\dots,e_{\alpha_n})^\top\big) = e_\alpha, \tag{2.10}$$

that is, the unit vector (vertex) of the meta-simplex $\Delta_N = \overline{\mathcal{S}_N}$ corresponding to the Dirac measure δ_α concentrated on the labeling $\alpha \in [c]^n$.

Example 2.1 We reconsider the toy scenario (1.6) of joint distributions of two binary variables. Such distributions correspond on the assignment manifold to points of the form

$$W = \Big(\big(\begin{smallmatrix} w_1 \\ 1-w_1 \end{smallmatrix}\big), \big(\begin{smallmatrix} w_2 \\ 1-w_2 \end{smallmatrix}\big) \Big)^\top, \qquad w_1, w_2 \in (0,1). \tag{2.11}$$

Embedding this point by (2.7) yields the vector

$$\begin{aligned} T(W) \; &= \big(w_1 w_2, w_1(1-w_2), (1-w_1)w_2, \\ &\quad (1-w_1)(1-w_2)\big)^\top, \end{aligned} \tag{2.12}$$

with components $T(W)_\alpha$ indexed by the four possible labeling $\alpha \in \{(1,1),(1,0),(0,1),(0,0)\}$. Since any distribution on the assignment manifold factorizes, this vector is determined by merely two parameters w_1, w_2. Accordingly, the embedded assignment manifold $\mathcal{T} = T(\mathcal{W}_c) \subset \mathcal{S}_N$ is the two-dimensional submanifold depicted by Fig. 1a.

In mathematics, such embedded sets are known as *Segre varieties* at the intersection of algebraic geometry and statistics [19, 38].

The following proposition highlights the specific role of the submanifold of $\mathcal{S}_N$ corresponding to the *embedded assignment manifold* $\mathcal{T} = T(\mathcal{W}_c) \subset \mathcal{S}_N$.

Proposition 2.2 ([5, Prop. 3.2]) *For every $W \in \mathcal{W}_c$, the distribution $T(W) \in \mathcal{S}_N$ has the maximum entropy*

$$H\big(T(W)\big) = -\sum_{\alpha\in[c]^n} T(W)_\alpha \log T(W)_\alpha \tag{2.13}$$

among all $p \in \mathcal{S}_N$ subject to the marginal constraint

$$Mp = W, \tag{2.14a}$$

where the marginalization map is given by

$$M: \mathbb{R}^N \to \mathbb{R}^{n\times c},$$

$$(Mp)_{i,j} := \sum_{\alpha\in[c]^n:\, \alpha_i = j} p_\alpha, \quad \forall (i,j) \in [n]\times[c]. \tag{2.14b}$$

As a consequence, any *general* distribution $p \in \mathcal{S}_N \setminus T(\mathcal{W}_c)$, which is *not* in $T(\mathcal{W}_c)$, has *non*-maximal entropy and hence is *more* informative by encoding additional statistical dependencies [14].

Our approach for *generating* such general distributions $p \in \mathcal{S}_N$, by combining simple factorizing distributions $W \in \mathcal{W}_c$ via the embedding (2.7) and assignment flows (2.3), is introduced in Sect. 3.

3 Approach

Section 3.1 introduces our generative model for representing and learning a discrete joint distribution $p = p(\alpha) \in \mathcal{S}_N$ of label configurations $\alpha = (\alpha_1,\dots,\alpha_n)$ as realizations of discrete random variables $y = (y_1,\dots,y_n) \sim p$. The approach is illustrated by Fig. 2. The training procedure for simulation-free training of the generative model is worked out in Sect. 3.2. Section 3.3 specifies precisely how the approximation of p is achieved in the meta-simplex by measuring transport on the embedded nonlinear submanifold of factorizing distributions.

We conclude with Sects. 3.4–3.6 on the geometric integration method that we employed for the discretization of our time-continuous generative model in numerical experiments, on the computation of the likelihood $\widetilde{p}(\alpha)$ of arbitrary label configurations using the learned generative model, and on the characterization of our approach as a dequantization procedure.

3.1 Generative Model

3.1.1 Goal

The goal is to learn an approximation

$$\widetilde{p} \approx p, \qquad \text{(approximation)} \tag{3.1}$$

as convex combination of *factorizing* joint distributions. The submanifold $\mathcal{T} = T(\mathcal{W}_c) \subset \mathcal{S}_N$ shown in Fig. 1a spans all factorizing distributions $T(W) \in \mathcal{S}_N$, which are efficiently represented by their marginals $W \in \mathcal{W}_c$ due to (2.14a). In particular, since the dimension of $\mathcal{W}_c$ only grows linearly in the number of variables n, factorizing distributions are tractable to work with numerically. However, only *independent* random variables follow factorizing distributions, posing the question of how statistical *coupling* between such variables can be represented through convex combination.

3.1.2 Representation of General Distributions

Note that the submanifold of factorizing distributions $\mathcal{T} \subseteq \mathcal{S}_N$ is *nonconvex*. Thus, convex combinations of two factorizing distributions $T(W_1)$ and $T(W_2)$ generally lie *outside* of $\mathcal{T}$ and hence form a *non-factorizing* distribution.

In addition, we observe that every Dirac measure e_α factorizes. Intuitively, this is because each variable has a deterministic value, independent of all others. Because Dirac measures are the extreme points of the convex set $\overline{\mathcal{S}_N}$, *every* joint distribution $\widetilde{p} \in \mathcal{S}_N$ representing an *arbitrary* coupling between variables can be written as a convex combination of Dirac measures

$$\widetilde{p} = \sum_{\alpha \in [c]^n} \widetilde{p}_\alpha e_\alpha. \tag{3.2}$$

This particular representation of $\widetilde{p}$ is intractable, however, because it involves a combinatorially large number of mixture coefficients $\widetilde{p}_\alpha$. To tame this complexity, the **key idea** is to *represent mixtures $\widetilde{p} \in \mathcal{S}_N$ of factorizing distributions as measures $\nu \in \mathcal{P}(\mathcal{W}_c)$* by

$$\widetilde{p} = \mathbb{E}_{W\sim\nu}[T(W)]. \tag{3.3}$$

This shifts the problem of parameterizing useful subsets of combinatorially many mixture coefficients in (3.2) to the problem of parameterizing a preferably large subset of measures $\nu \in \mathcal{P}(\mathcal{W}_c)$, supported on the comparatively low-dimensional manifold $\mathcal{W}_c$. The latter can be achieved by *parameterized measure transport* on the *assignment manifold $\mathcal{W}_c$*.

Specifically, a simple reference measure

$$\nu_0 \in \mathcal{P}(\mathcal{W}_c) \qquad \text{(reference measure)} \tag{3.4a}$$

is chosen and transported by the assignment flow (2.3), reaching

$$\nu = \nu_\infty \quad \text{for} \quad t \to \infty. \qquad \text{(transported measure)} \tag{3.4b}$$

Parameterization of measures

$$\nu_\theta \in \mathcal{P}(\mathcal{W}_c) \qquad \text{(parametrized measure)} \tag{3.4c}$$

is achieved by choosing an appropriate class of affinity functions $F_\theta \colon \mathcal{W}_c \to \mathbb{R}^{n\times c}$ (1.12) driving the assignment flow (2.3). Note that, while the support of $\widetilde{p}$ in (3.2) was directly associated with the number of mixture coefficients, the complexity of representing $\widetilde{p}$ via the ansatz (3.3) is no longer associated with its support.

The simplest example of (3.3) is the representation of:

$$\widetilde{p} = \mathbb{1}_{\mathcal{S}_N} = \frac{1}{N}\mathbb{1}_N \tag{3.5}$$

by choosing $F_\theta \equiv 0$ and a *product* reference distribution

$$\nu_0 = \prod_{i\in[n]} \nu_{0;i} \in \mathcal{P}(\mathcal{W}_c) \tag{3.6}$$

with mean $\mathbb{E}_{W_i\sim\nu_{0;i}}[W_i] = \mathbb{1}_{\mathcal{S}}$, which through the embedding (3.3) yields (3.5), which has *full* support on the very high-dimensional space $[c]^n$. We make this connection more explicit.

Lemma 3.1 (convex combination of embedded nodewise measures) *Suppose the reference measure ν_0 has the product form (3.6) with $\nu_i \in \mathcal{P}(\mathcal{S}_c)$. Then, the joint distribution represented by the mixture (3.3) reads:*

$$\begin{aligned} \widetilde{p} &= \mathbb{E}_{W\sim\nu}[T(W)] = T(\widehat{W}), \\ \widehat{W}_i &= \mathbb{E}_{W_i\sim\nu_i}[W_i], \quad i \in [n]. \end{aligned} \tag{3.7}$$

Proof Let $\alpha \in [c]^n$ be an arbitrary multi-index. Since ν factorizes in the described manner, $W \sim \nu$ is independently distributed on each node which implies

$$\begin{aligned} \widetilde{p}_\alpha &= \mathbb{E}_{W\sim\nu}[T(W)_\alpha] = \mathbb{E}_{W\sim\nu}\Big[\prod_{i\in[n]} W_{i,\alpha_i}\Big] \\ &= \prod_{i\in[n]} \mathbb{E}_{W\sim\nu_i}[W_{i,\alpha_i}] = T(\widehat{W})_\alpha. \end{aligned} \tag{3.8}$$

□

Lemma 3.1 shows that, if ν is independent on every node, then $\widetilde{p} \in \mathcal{T}$. In particular, *coupling* between variables, to be represented by the joint distribution $\widetilde{p}$, has necessarily to be induced by the *interaction* of node states over the course of integrating the assignment flow.

3.1.3 Model Learning and Model Evaluation (Sampling)

The target distribution p is unknown, in practice, and only independently drawn training samples $\beta \sim p$ are available. After choosing a class of payoff functions F_θ, the task is to learn parameters θ such that

$$\widetilde{p} = \mathbb{E}_{W\sim\nu_\theta}[T(W)], \tag{3.9}$$

i.e., a parametrization of the right-hand side of (3.3), approximates the empirical distribution of samples. To this end, we identify samples β with the corresponding extremal points $Me_\beta \in \overline{\mathcal{W}_c}$ (Sect. 3.2.1) and use *flow matching* on $\mathcal{W}_c$ to learn θ in a numerically stable and efficient way (Sect. 3.2).

After learning has converged, new samples $\beta \sim \widetilde{p}$ from the approximate distribution $\widetilde{p} \approx p$ can be drawn by a two-stage process:

(i) First, an initialization $W_0 \sim \nu_0$ is drawn and evolved over time $W(t) \in \mathcal{W}_c$ by integrating the learned assignment flow until either the desired time $t_{\max}$ is reached, or $W(t)$ approaches an extreme point of $\overline{\mathcal{W}_c}$.

(ii) The new data is subsequently drawn from the factorizing distribution $T(W(t_{\max}))$. At extreme points $Me_{\beta'}$, this distribution is a Dirac measure and sampling from it always yields β'.

3.2 Riemannian Flow Matching

In this section, we work out details of the procedure for training generative assignment flows.

3.2.1 Representation of Labelings as Training Data

Our approach to training the generative model utilizes *labelings* as training data of the form

$$\overline{W} \in \overline{\mathcal{W}_c}, \qquad \overline{W}_i = e_{\alpha_i}, \qquad \alpha_i \in [c], \qquad \forall i \in [n]. \tag{3.10}$$

Any such point $\overline{W}$ assigns a label (category) α_i to each vertex $i \in \mathcal{V}$ in terms of a corresponding unit vector $e_{\alpha_i} \in \{0,1\}^c$. The flow-matching criterion, specified in the following section, is optimized to find θ such that F_θ drives the assignment flow to labelings in the limit $\lim_{t\to\infty} W(t) = \overline{W}$. In practice, the assignment flow is integrated up to a sufficiently large point of time

$$t_{\max} > 0 \tag{3.11}$$

followed by trivial rounding of $W_i(t_{\max}) \mapsto e_{\alpha_i}$ at every vertex i.

3.2.2 Training Criterion

This section details the approach schematically depicted by Fig. 2. In the following,

$$\beta \sim p \tag{3.12}$$

denotes labeling configurations for training, drawn from the *unknown* underlying discrete joint data distribution p. β corresponds to the Dirac measure $e_\beta \in \mathcal{S}_N$ (extreme point) of the meta-simplex $\mathcal{S}_N$ and to a corresponding point $\overline{W}_\beta = Me_\beta \in \overline{\mathcal{W}_c}$, to which the assignment flow (2.3) may converge.

The idea of flow matching is to directly fit the model vector field, in our case the assignment flow vector field (2.3),

$$V_\theta(W,t) := R_W[F_\theta(W,t)], \tag{3.13}$$

to a vector field whose flow realizes a desired measure transport. Let $\nu_0 \in \mathcal{P}(\mathcal{W}_c)$ be a simple reference measure and define *conditional probability paths*

$$\nu_t(\beta) \tag{3.14}$$

satisfying

$$\nu_0(\beta) := \nu_0 \tag{3.15a}$$

$$\nu_\infty(\beta) := \lim_{t\to\infty} \nu_t(\beta) = \delta_{\overline{W}_\beta}(W) \quad \text{for all} \quad \beta \in [c]^n, \tag{3.15b}$$

where $\overline{W}_\beta = (e_{\beta_1},\dots,e_{\beta_n})^\top \in \{0,1\}^{n\times c}$ is the extreme point of $\overline{\mathcal{W}_c}$ corresponding to β, such that $T(\overline{W}_\beta) = e_\beta \in \mathcal{S}_N$. Then, the *marginal probability path*

$$\nu_t = \mathbb{E}_{\beta\sim p}[\nu_t(\beta)] \tag{3.16}$$

represents the target data distribution p in the limit $t \to \infty$ by ν_∞ and

$$\mathbb{E}_{W\sim\nu_\infty}[T(W)] = \mathbb{E}_{\beta\sim p}[e_\beta] = p. \tag{3.17}$$

In principle, we can now define a vector field

$$u_t \colon \mathcal{W}_c \to \mathcal{T}_0 \tag{3.18}$$

which generates the path $t \mapsto \nu_t$ in the sense that the flow of u_t pushes forward ν_0 to ν_t, for all times $t \geq 0$. Let $\rho \in \mathcal{P}([0,\infty))$ be a distribution with full support on the nonnegative time axis. Regression of the assignment flow vector field (3.13),

$$V_\theta(\cdot,t) \colon \mathcal{W}_c \to \mathcal{T}_0, \tag{3.19}$$

with respect to u_t, amounts to minimizing the **Riemannian flow-matching criterion**

$$\mathcal{L}_{\mathrm{RFM}}(\theta) = \mathbb{E}_{t\sim\rho,W\sim\nu_t}\Big[\big\|u_t(W) - V_\theta(W,t)\big\|_W^2\Big], \tag{3.20}$$

where $\|\cdot\|_W^2 = g_W(\cdot,\cdot)$ (cf. (2.2e)).

In this form, flow matching is intractable, however, because we do not have access to the required field u_t. On the other hand, since we are at liberty to define *conditional* paths that conform to the constraints (3.15), we can choose $\nu_t(\beta)$ that are generated by *conditional* vector fields $u_t(\cdot|\beta)$ with the *known* form. The key insight in [11], based on [36] and provided that each $\nu_t(\beta)$ is generated by $u_t(\cdot|\beta)$, is that the loss function (3.20) has the same gradient with respect to θ as the **Riemannian *conditional* flow-matching criterion**

$$\mathcal{L}_{\mathrm{RCFM}}(\theta) = \mathbb{E}_{t\sim\rho,\beta\sim p,W\sim\nu_t(\beta)}\Big[\big\|u_t(W|\beta) - V_\theta(W,t)\big\|_W^2\Big] \tag{3.21a}$$

$$\overset{(3.13)}{=} \mathbb{E}_{t\sim\rho,\beta\sim p,W\sim\nu_t(\beta)}\Big[\big\|u_t(W|\beta) - R_W[F_\theta(W,t)]\big\|_W^2\Big]. \tag{3.21b}$$

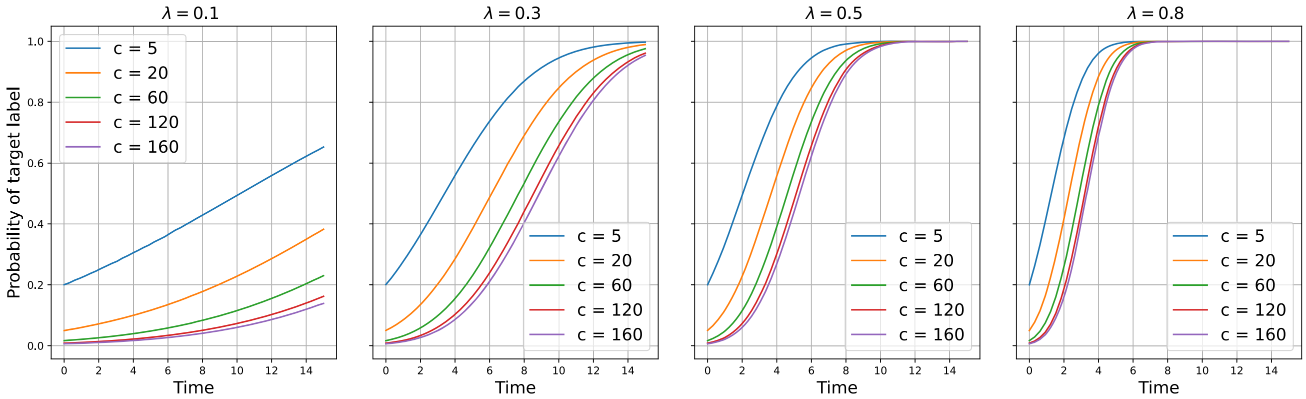


Fig. 3 Influence of the parameter λ controlling in (3.26) and (3.30), respectively, the *rate* of assignment of mass of the pushforward probability measure (3.27) to a target label, depending on the number c of labels (classes, categories)

By contrast to (3.20), conditional vector fields $u_t(W|\beta)$ generating a path

$$t \mapsto \nu_t(\beta) \tag{3.22}$$

with the required properties (3.15) can be specified in a closed form (cf. Proposition 3.28 below), and the conditional loss function (3.21) can be evaluated efficiently. Ultimately, by minimizing (3.21), the measure ν_t generated from the reference measure ν_0 by the assignment flow vector field $R_W[F_\theta(W,t)]$ approximates ν_∞ in the limit $t \to \infty$, which represents the unknown data distribution p through (3.17).

3.2.3 Constructing Conditional Fields

This section specifies the conditional vector fields $u_t(W|\beta)$ that generate the paths (3.22) conforming to (3.15) and define the conditional flow-matching objective (3.21).

Let

$$\mathcal{N}_0(V) := \mathcal{N}(V; 0, \Pi_0) \tag{3.23}$$

denote the standard Gaussian centered in the tangent space at $0 \in \mathcal{T}_0$, with the orthogonal projection (2.2d) representing the identity map on $\mathcal{T}_0 \subset \mathbb{R}^{n\times c}$. Pushing forward $\mathcal{N}_0$ by the lifting map (2.6b) at the barycenter yields a simple *reference distribution*

$$\nu_0 = (\exp_{\mathbb{1}_\mathcal{W}})_\sharp \mathcal{N}_0 \in \mathcal{P}(\mathcal{W}_c). \tag{3.24}$$

The distribution (3.24) is *simple* in the sense that it is easy to draw samples and the conditions of Lemma 3.1 are satisfied; in particular, ν_0 *factorizes* nodewise. For each labeling $\beta \in [c]^n$ and the corresponding extreme point $\overline{W}_\beta = (e_{\beta_1}, \dots, e_{\beta_n}) \in \overline{\mathcal{W}_c}$, and a

$$\lambda > 0, \qquad \text{(rate parameter)} \tag{3.25}$$

define the probability path

$$t \mapsto \mathcal{N}_{t,\beta} := \mathcal{N}(\cdot; t\lambda V_\beta, \Pi_0) \in \mathcal{P}(\mathcal{T}_0), \qquad V_\beta := \Pi_0 \overline{W}_\beta, \tag{3.26}$$

and lift it to $\mathcal{W}_c$, defining

$$\nu_t(\beta) := (\exp_{\mathbb{1}_\mathcal{W}})_\sharp \mathcal{N}_{t,\beta}. \tag{3.27}$$

The parameter λ controls the *rate* at which $\nu_t(\beta)$ moves probability mass closer to $\overline{W}_\beta$. Small values of λ move the mass slowly; this is useful in settings with many labels $c \gg 1$, enabling the process to make class decisions during a longer time period. Figure 3 illustrates quantitatively the influence of λ.

The following proposition makes explicit the conditional vector field $u_t(W|\beta)$ that generates (3.27) and hence defines the training objective function (3.21). Recall the notation of Sect. 2.2 and the first paragraph of Sect. 3.2.2 explaining the one-to-one correspondence between

- a labeling configuration β,
- the corresponding Dirac measure $e_\beta \in \mathcal{S}_N$ of the meta-simplex, and
- the corresponding point $\overline{W}_\beta \in \overline{\mathcal{W}_c}$ of the closure of the assignment manifold.

Proposition 3.2 (conditional vector fields) *The probability paths defined in* (3.27) *are generated through the smooth flow*

$$\psi_\cdot(\cdot|\beta)\colon \mathbb{R}_{\geq 0} \times \mathcal{T}_0 \to \mathcal{W}_c, \quad \psi_t(V|\beta) = \exp_{\mathbb{1}_\mathcal{W}}(V + t\lambda V_\beta). \tag{3.28}$$

It is invertible and has the smooth inverse

$$\psi_t^{-1}(W|\beta) = \exp_{\mathbb{1}_\mathcal{W}}^{-1}(W) - t\lambda V_\beta. \tag{3.29}$$

In particular, the conditional vector field that generates (3.27) *is given by*

$$u_t(W|\beta) = R_W[\lambda V_\beta]. \tag{3.30}$$

Proof See Appendix A.1, page 18. □

Proposition 3.3 (conditional path constraints) *The conditional probability paths $\nu_t(\beta)$ defined by (3.27) satisfy the constraints (3.15).*

Proof See Appendix A.1, page 19. □

The path $\mathcal{N}_t$ is generated on the tangent space $\mathcal{T}_0$ by the constant vector field $V \mapsto \lambda V_\beta$ given by (3.26). The related vector field on $\mathcal{W}_c$, which generates the path (3.27), is given by (3.30). Comparing the shape of this field to (2.3) makes clear that assignment flows are natural candidate dynamics for matching conditional paths of the described form. The Riemannian conditional flow-matching objective (3.21) consequently reads:

$$\mathcal{L}_{\mathrm{RCFM}}(\theta) = \mathbb{E}_{t\sim\rho,\beta\sim p,W\sim\nu_t(\beta)}\Big[\big\|R_W[\lambda V_\beta - F_\theta(W,t)]\big\|_W^2\Big]. \tag{3.31}$$

We point out that this criterion is ‘simulation-free’, i.e., *no integration* of the assignment flow is required for loss evaluation, which makes training computationally efficient.

Our approach (3.31) constitutes a novel instance of the flow-matching approach to generative modeling, introduced by [36] and recently extended to Riemannian manifolds by [11]. This instance uses the assignment manifold (1.9) and the corresponding Riemannian flow (2.3), along with the meta-simplex embedding (2.7), to devise a generative model whose underlying information geometry tailors the model to the representation and learning of *discrete joint* probability distributions.

3.2.4 Infinite Integration Time

A notable difference between our approach and previous Riemannian flow-matching methods is that the target distribution is reached for $t \to \infty$ rather than after finite time. This corresponds to the fact that e-geodesics do not reach boundary points of $\overline{\mathcal{W}_c}$ after finite time and thus avoids two problems faced in prior work.

First, unlike the preliminary version presented in [6], data points $\beta \in [c]^n$ do not need to be smoothed in order to present targets in the interior of $\mathcal{W}_c$. Instead, we can directly approach extreme points $\overline{W}_\beta \in \overline{\mathcal{W}_c}$, even though they are at infinity in the tangent space $\mathcal{T}_0$ at $\mathbb{1}_{\mathcal{W}}$. Figure 4 shows that working within a ball in T_0 with radius 15 suffices to represent ‘infinity’ in practice.

Second, by not moving all mass of the reference distribution (close) to $\overline{W}_\beta$ in finite time, we avoid a pathological behavior which can arise in flow matching on the simplex. Denote by

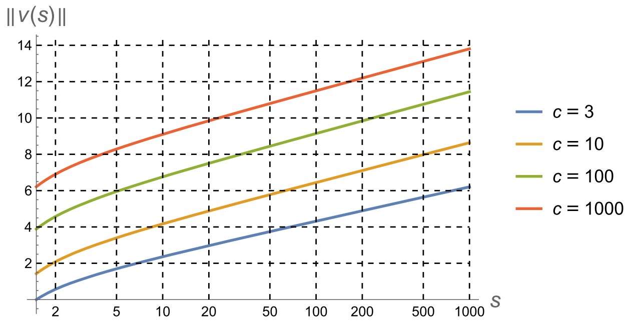


Fig. 4 Norms $\|v(s)\|$ of the tangent vectors $v(s) = \exp_{\mathbb{1}_{\mathcal{S}}}^{-1}\big(p(s)\big)$ with $p(s) = (\frac{s-1}{s}, \frac{1}{(c-1)s}, \dots, \frac{1}{(c-1)s}) \to e_1 \in \mathbb{R}^c$ if $s \to \infty$, for numbers of labels $c \in \{3, 10, 100, 1000\}$. Since $\|e_1 - p(s)\| = \big(\frac{c}{c-1}\big)^{1/2}\frac{1}{s} \approx \frac{1}{s}$, the simplex Δ_c is covered, up to a very small distance to its boundary, by $\exp_{\mathbb{1}_{\mathcal{S}}}(B_0(r)) \subset \mathcal{S}_c$ and tangent vectors $v \in B_0(r) \subset T_0$ within a ball $B_0(r)$ centered at $0 \in T_0$ with radius $r = 15$

$$r_\beta = \Big\{W \in \mathcal{W}_c \colon \beta_i \in \arg\max_{j\in[c]} W_{i,j},\ \forall i \in [n]\Big\} \tag{3.32}$$

the subset of points in $\mathcal{W}_c$ which assign their largest probability to the labels β. [42, Proposition 1] lays out that moving all mass of the reference distribution (close) to $\overline{W}_\beta$ in *finite* time forces the model to make class decisions very early because the probability of r_β under $\nu_t(\beta)$ increases too quickly. The effect is exacerbated by increasing the number of classes c that the model is asked to discriminate.

However, by opting for large integration time $t \to \infty$ and a corresponding construction (3.27) of conditional probability paths, our approach is able to scale to many classes $c \gg 1$, avoiding the pathology described in [42, Proposition 1]. Formally, this is because $\nu_t(\beta)$ defined in (3.27) has full support on $\mathcal{W}_c$ for every $t \geq 0$. In practice, the parameter λ in (3.26) can be used to control the speed at which the probability of r_β under $\nu_t(\beta)$ increases, allowing the model to make class decisions gradually over time.

Figure 9 (page 17) displays probability density paths for illustration. The corresponding impact on model accuracy is quantitatively shown in Fig. 5 (page 12), with experimental details elaborated in Sect. 4.1.

3.2.5 Relation to Dirichlet Flow Matching

The construction of [42] specifically addresses pathological behavior of flow matching on the simplex, by choosing conditional probability paths $\nu_t(\beta)$ as paths of Dirichlet distributions. They demonstrate that this approach scales to at least $c = 160$ classes, by allowing the model to make class decisions gradually over time. However, the explicit definition of $\nu_t(\beta)$ as paths of Dirichlet distributions makes it

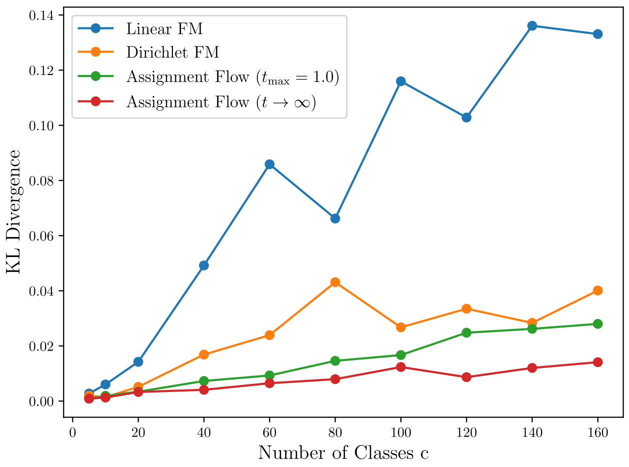


Fig. 5 Relative entropy between learned models (histogram of 512k samples) and a known, factorizing target distribution on $n = 4$ simplices with varying number of classes c. By leveraging information geometry and gradual decision-making over time, our proposed approach (red) is able to outperform our earlier method [6] as well as Dirichlet flow matching [42] in terms of scaling to many classes c

non-trivial to find corresponding vector fields $u_t(\cdot|\beta)$ for flow matching, which leads them to make an ansatz for fields, which move mass along straight lines in the ambient Euclidean space in which the probability simplex is embedded.

While we also make an explicit choice for $\nu_t(\beta)$ in (3.27), our construction is notably simpler than the approach of [42], allowing to easily compute the vector fields $u_t(\cdot|\beta)$ by push-forward (Proposition 3.2). The resulting flow moves mass along e-geodesics on $\mathcal{W}_c$, which is much more natural with respect to the information geometry of discrete probability distributions. To illustrate this point, consider a straight path $\widehat{p}(t) \in \mathbb{R}^n$ with direction $\frac{d}{dt}\widehat{p}(t) = v \in \mathbb{R}^n$ at all times t. The trajectory $\widehat{p}(t)$ is generated by maximizing $\langle v, \widehat{p}\rangle$ along its gradient direction. On $\mathcal{W}_c$, the quantity $\langle V_\beta, W\rangle$ can be interpreted as correlation between $W \in \mathcal{W}_c$ and the direction V_β. The Riemannian gradient of this correlation with respect to the product Fisher–Rao geometry on $\mathcal{W}_c$ is $R_W[V_\beta]$, i.e., precisely the direction of the conditional vector field (3.30).

3.3 Learning Interaction Between Simplices

Our prior work [5] has studied the relationship between assignment flows on the product manifold $\mathcal{W}_c$ and replicator dynamics on the meta-simplex $\mathcal{S}_N$. We now use core results of [5] to derive the flow-matching approach of Sect. 3.2 from *first principles of flow matching in* $\mathcal{S}_N$, that is in the combinatorially large space of *all* discrete joint distributions. This demonstrates, in particular, that the proposed approach is suitable for *structured prediction* settings, in which multiple *coupled* random variables are of interest.

The result is surprising because *direct* flow matching of joint distributions in $\mathcal{S}_N$ is *intractable* due to the combinatorial dimension $N = c^n$. However, by leveraging the submanifold $\mathcal{T}$ (defined by (1.13) and illustrated by Fig. 1) and the compatibility of assignment flows with its geometry, we show that our construction can effectively break down combinatorial complexity and define a *numerically tractable method*.

The map $T : \mathcal{W}_c \to \mathcal{S}_N$ defined in (2.7) associates a marginal distribution of n discrete random variables $W \in \mathcal{W}_c$ with a factorizing joint distribution $T(W) \in \mathcal{S}_N$. Define with slight abuse of notation[1] the orthogonal projection

$$\pi_0 : \mathbb{R}^N \to \mathcal{T}_0\mathcal{S}_N \tag{3.33}$$

and formally denote the scaled standard normal distribution on $\mathcal{T}_0\mathcal{S}_N$ with variance c^{n-1} by

$$\begin{aligned}\mathcal{N}_0^{\mathcal{S}_N} &= (\sqrt{c^{n-1}}\pi_0)_\sharp \mathcal{N}(0, I_N) = \mathcal{N}(0, c^{n-1}\pi_0\pi_0^\top) \\ &= \mathcal{N}(0, c^{n-1}\pi_0).\end{aligned} \tag{3.34}$$

Analogous to the construction of conditional measures in Sect. 3.2.3, we define the path of conditional measures

$$\mathcal{N}_t^{\mathcal{S}_N}(\cdot|\beta) = \mathcal{N}(\cdot; tc^{n-1}\lambda\pi_0 e_\beta, c^{n-1}\pi_0) \tag{3.35}$$

given a labeling $\beta \in [c]^n$ and a rate parameter $\lambda > 0$, scaled by the constant c^{n-1}. It follows from Proposition 3.3 that

$$\nu_t^{\mathcal{S}_N}(\beta) = (\exp_{\mathbb{1}_{\mathcal{S}_N}})_\sharp \mathcal{N}_t^{\mathcal{S}_N}(\cdot|\beta) \tag{3.36}$$

satisfies the conditions (3.15) on $\mathcal{S}_N$ and is thus suitable for flow matching on $\mathcal{S}_N$ with reference distribution $\nu_0^{\mathcal{S}_N} = \mathcal{N}_0^{\mathcal{S}_N}$. Formally, the Riemannian conditional flow-matching criterion analogous to (3.31) reads:

$$\begin{aligned}\mathcal{L}_{\mathrm{RCFM}}^{\mathcal{S}_N}(\theta) = \mathbb{E}&_{t\sim\rho,\beta\sim p,q\sim\nu_t^{\mathcal{S}_N}(\beta)} \\ &\Big[\big\| R_q[\lambda\pi_0 e_\beta - f_\theta(q,t)]\big\|_w^2\Big]\end{aligned} \tag{3.37}$$

for an affinity function $f_\theta : \mathcal{S}_N \times [0, \infty) \to \mathcal{T}_0\mathcal{S}_N$.

The task of minimizing (3.37) is numerically intractable, because we are not even able to easily represent general *points* $q \in \mathcal{S}_N \setminus \mathcal{T}$ in the complement of the embedded assignment manifold $\mathcal{T} = T(\mathcal{W}_c)$ given by (2.7). To break down this complexity, we will define a projection onto $\mathcal{T}$ by using the *lifting map lemma* [5, Lemma 3.3], which states

$$\exp_{\mathbb{1}_{\mathcal{S}_N}}(QV) = T\big(\exp_{\mathbb{1}_{\mathcal{W}}}(V)\big) \tag{3.38}$$

[1] π_0 is defined by (2.1d) as orthogonal projection onto the tangent space $T_0\mathcal{S}_c$ of the *single* simplex $\mathcal{S}_c$ with trivial tangent bundle $\mathcal{S}_c \times T_0$. Here, to simplify notation, we overload π_0 to denote analogously the orthogonal projection onto the tangent space $\mathcal{T}_0\mathcal{S}_N$.

for all tangent vectors $V \in \mathcal{T}_0$, with the mappings T and Q defined by (2.7) and (2.9). We start with an orthogonal projection $\mathcal{T}_0\mathcal{S}_N \to \operatorname{img}(Q) \cap \mathcal{T}_0\mathcal{S}_N$.

Lemma 3.4 (orthogonal projection onto $\operatorname{img}(Q) \cap \mathcal{T}_0\mathcal{S}_N$) *The orthogonal projection* proj_0 *of tangent vectors in* $\mathcal{T}_0\mathcal{S}_N$ *to the subspace* $\operatorname{img} Q \cap \mathcal{T}_0\mathcal{S}_N$ *reads*

$$\begin{aligned} &\operatorname{proj}_0 : \mathcal{T}_0\mathcal{S}_N \to \operatorname{img} Q \cap \mathcal{T}_0\mathcal{S}_N, \\ &\quad \operatorname{proj}_0(v) := Q_c \Pi_0 Q_c{}^\top v, \qquad \textit{for } v \in \mathcal{T}_0\mathcal{S}_N, \end{aligned} \tag{3.39}$$

in terms of the linear operator

$$Q_c := \frac{1}{\sqrt{c^{n-1}}} Q. \tag{3.40}$$

Since (3.38) ensures that $\exp_{\mathbb{1}_{\mathcal{S}_N}}(\operatorname{img} Q) \subseteq \mathcal{T}$, we can now define the projection

$$\operatorname{proj}_{\mathcal{T}} := \exp_{\mathbb{1}_{\mathcal{S}_N}} \circ \operatorname{proj}_0 \circ \exp^{-1}_{\mathbb{1}_{\mathcal{S}_N}} : \mathcal{S}_N \to \mathcal{T}. \tag{3.41}$$

Under this projection, the conditional measures $\nu_t^{\mathcal{S}_N}(\beta) \in \mathcal{P}(\mathcal{S}_N)$ precisely induce the conditional probability paths $\nu_t(\beta) \in \mathcal{P}(\mathcal{W}_c)$ defined by (3.27). Note that every extreme point of $\mathcal{S}_N$ lies in (the closure of) $\mathcal{T}$. Thus, projecting to $\mathcal{T}$ preserves the Dirac measures δ_{e_β} reached by the conditional distributions (3.36) in the limit $t \to \infty$. In particular, the projection transforms the intractable conditional flow-matching criterion (3.37) on $\mathcal{S}_N$ into the numerically tractable criterion (3.31).

Theorem 3.5 (projected flow matching on $\mathcal{S}_N$) *For any* $\beta \in [c]^n$, *the pushforward of the conditional measure* $\nu_t^{\mathcal{S}_N}(\beta)$ *defined in* (3.36) *under the projection* $\operatorname{proj}_{\mathcal{T}} \colon \mathcal{S}_N \to \mathcal{T}$ *defined in* (3.41) *is*

$$(\operatorname{proj}_{\mathcal{T}})_\sharp \nu_t^{\mathcal{S}_N}(\beta) = T_\sharp \nu_t(\beta) \tag{3.42}$$

with $\nu_t(\beta) \in \mathcal{P}(\mathcal{W}_c)$ *defined in* (3.27) *and the embedding map* T *given by* (2.7). *Furthermore, the flow-matching criterion on* $\mathcal{T}$, *induced by the conditional paths* (3.42), *reads*

$$\begin{aligned} \mathcal{L}^{\mathcal{T}}_{RCFM}(\theta) &= \mathbb{E}_{t\sim\rho,\beta\sim p,q\sim(proj_{\mathcal{T}})_\sharp \nu_t^{\mathcal{S}_N}(\beta)} \\ &\quad \Big[\big\| R_q[\lambda \pi_0 e_\beta - \widetilde{f}_\theta(q,t)] \big\|_w^2 \Big] \end{aligned} \tag{3.43}$$

and, using the ansatz $\widetilde{f}_\theta = Q \circ F_\theta \circ M$ *with* Q *and* M *defined by* (2.9) *and* (2.14b), (3.43) *is equal to the criterion* (3.31) *for matching assignment flows on* $\mathcal{W}_c$.

Theorem 3.5 shows that the constructed flow matching on $\mathcal{W}_c$, which operates separately on multiple simplices, is *induced* by flow matching in the *single* meta-simplex $\mathcal{S}_N$, with conditional distribution paths and vector fields projected to the submanifold $\mathcal{T} = T(\mathcal{W}_c)$.

This result provides a geometric justification of the fact that *interaction* between simplices is learned through flow matching, even though all conditional probability paths $\nu_t(\beta)$ used for training can be *separately* constructed on individual simplices.

3.4 Numerical Flow Integration

We point out again that learning our generative model by Riemannian flow matching is 'simulation-free': numerical integration is not required since only vector fields have to be matched, which are defined on the tangent bundle of the assignment manifold and on the corresponding tangent-subspace distribution of the meta-simplex (Prop. 3.5), respectively. On the other hand, numerical integration of the flow is required for evaluating the learned generative model, in order to sample as illustrated by Fig. 1, or for likelihood computation (Sect. 3.5).

Since the flow corresponds to an ODE on a Riemannian manifold, *geometric* numerical integration utilizes various representations of the ODE on the tangent bundle in order to apply established methods for numerical integration on Euclidean spaces [25]. In the case of the assignment flow, this has been thoroughly studied by [48] using the extension of the lifting map (2.6a) to the product manifold (2.6b), regarded as action of the respective tangent space (regarded as additive abelian Lie group) on the assignment manifold. From the general viewpoint of geometric numerical integration, the resulting schemes for geometric numerical integration categorize as Runge–Kutta schemes of Munthe–Kaas type [39].

Specifically, in this paper, numerical integration was carried out using the classical explicit embedded Dormand & Prince Runge–Kutta method [17] of orders 4 & 5 with step-size control (cf. [48, Section 5.2] and [27, Section II.5]).

3.5 Likelihood Computation

The likelihood of test data under the model distribution $\widetilde{p}$ is commonly used as a surrogate for Kullback–Leibler divergence between $\widetilde{p}$ and the true data distribution p, due to the relationship

$$\mathrm{KL}(p, \widetilde{p}) = \mathbb{E}_p\Big[\log \frac{p}{\widetilde{p}}\Big] = -H(p) - \mathbb{E}_p[\log \widetilde{p}]. \tag{3.44}$$

The entropy $H(p)$ is a property of the data distribution, which is not typically known, but can be treated as a constant which does not depend on the model used to approximate $\widetilde{p}$. For continuous normalizing flows, likelihood under the model is directly used as a training criterion, for this reason. Using the

Table 1 Likelihood of binarized MNIST test data under our proposed model ($t \to \infty$) and the earlier version [6] ($t \to 1$)

Method	AF ($t \to \infty$)	AF ($t \to 1$)
Likelihood (bits / dim)	1.01 ± 0.17	4.05 ± 0.83

Both methods are trained by flow matching rather than likelihood maximization

instantaneous change-of-variables formula [13]

$$\frac{\partial}{\partial t} \log \nu_t(x) = -\operatorname{tr} J(x,t), \tag{3.45}$$

log-likelihood under continuous normalizing flows can, on continuous state spaces, be computed by integrating (3.45) backward in time. In (3.45), $J(x,t)$ denotes the vector field Jacobian, whose trace is commonly approximated by using Hutchinson's estimator [29]

$$\operatorname{tr} J = \mathbb{E}_v[\langle v, Jv \rangle] \tag{3.46}$$

with v drawn from a fixed normal or Rademacher distribution. The use of this estimator in the context of likelihood under continuous normalizing flows was proposed by [21]. The authors use a single sample v for each integration of (3.45), which yields an unbiased estimator for log-likelihood of independent test data. In order to use likelihood as a training criterion, numerical integration of (3.45) is required. This entails many forward and backward passes through the employed network architecture in order to compute a single parameter update.

Therefore, we do not use likelihood as a training criterion, opting instead for the simulation-free flow-matching approach of Sect. 3.2. Since the learned model is still a normalizing flow, (3.45) remains a useful tool for computing likelihoods under our model. However, because we are modeling discrete data while working on continuous state spaces, likelihood of discrete data cannot be computed as a point estimate on $\mathcal{W}_c$. Further details are provided in Appendix C.

3.6 Dequantization

Approximation of discrete data distributions by continuous distributions has been studied through the lens of *dequantization*. Choose a latent space $\mathcal{F}^n$ and an embedding of class label configurations $\beta \in [c]^n$ as prototypical points $f^*_\beta \in \mathcal{F}^n$. Suppose the choice of these points is fixed before training and associate disjoint sets $A_\beta \subseteq \mathcal{F}^n$ with label configurations such that they form a partition of $\mathcal{F}^n$ and $f^*_\beta \in A_\beta$. We can then define the continuous surrogate model

$$\vartheta = \sum_{\beta \in [c]^n} p_\beta \mathcal{U}_{A_\beta} \in \mathcal{P}(\mathcal{F}^n) \tag{3.47}$$

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

$$\mathbb{P}_\vartheta(A_\beta) = \int_{A_\beta} \vartheta(y)dy = p_\beta \int_{A_\beta} \mathcal{U}_{A_\beta}(y)dy = p_\beta \tag{3.48}$$

due to the disjoint support of mixture components in (3.47). Denote a continuous model distribution by $\nu \in \mathcal{P}(\mathcal{F}^n)$. Using Jensen's inequality, we find

$$\begin{aligned} -H(\vartheta) - \mathrm{KL}(\vartheta, \nu) &= \int \vartheta(y) \log \nu(y) dy \\ &= \sum_{\beta \in [c]^n} p_\beta \int_{A_\beta} \log \nu(y) dy && (3.49a) \\ &\leq \sum_{\beta \in [c]^n} p_\beta \log \int_{A_\beta} \nu(y) dy && (3.49b) \\ &= -H(p) - \mathrm{KL}(p, q) && (3.49c) \end{aligned}$$

for the discrete model distribution q defined by

$$q_\beta = \int_{A_\beta} \nu(y) dy = \mathbb{P}_\nu(A_\beta). \tag{3.50}$$

Thus, fitting ν to ϑ by maximizing log-likelihood of smoothed data drawn from ϑ 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 $f^*_{\beta_k} \in \mathcal{F}^n$ of discrete data $\{\beta_k\}_{k \in [n]}$.

The above *dequantization approach* was first proposed by [46]. Their reasoning justifies the previously known heuristic of adding noise to dequantize data [47]. It has thenceforth become common practice for training normalizing flows as generative models of images [18, 43] and was generalized to non-uniform noise distributions by [24]. These 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 $\mathcal{F}^n$.

For the *general* discrete data considered here, such a structure is not available. As a remedy, [9] present an approach to learn the embedding jointly with likelihood maximization and defining the partition of $\mathcal{F}^n$ into subsets A_β through Voronoi tessellation. The rounding model variant (C.1) of our approach can be seen as dequantization on the space $\mathcal{F}^n = \mathcal{W}_c$ with prototypical points $f^*_\beta = \overline{W}_\beta$. The sets A_β generated by Voronoi tessellation then coincide with the sets r_β defined by (3.32). However, our approach differs from [9] by using flow matching instead of likelihood-based training and by explicit consideration of information geometry on $\mathcal{W}_c$.

A natural question is whether the ability to learn an embedding of class configurations as prototypical points f^*_β, thereby representing similarity relations between classes, can be replicated in our setting. Indeed, because points in $\mathcal{S}_c$ have a clear interpretation as categorical distributions, it is easy to achieve this goal by extending the affinity function F_θ of the assignment flow (2.3).

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 an integer probability vector $e_j \in \mathcal{S}_c$ precisely selects one of these points, 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 nodewise. We now choose a parameterized function $\widetilde{F}_\theta\colon \mathbb{R}^L \to \mathbb{R}^L$ that operates on $\mathbb{R}^L$ and define the extended payoff function

$$F_\theta = \mathcal{E}^\top \circ \widetilde{F}_\theta \circ \mathcal{E}\colon \mathcal{W}_c \to \mathbb{R}^{n\times c}. \tag{3.51}$$

4 Experiments and Discussion

As outlined in Sect. 3, we perform Riemannian flow matching (3.20) via the conditional objective (3.31) to learn assignment flows (2.3). These in turn approximate p_∞ in the limit $t \to \infty$ and thereby the unknown data distribution p through (3.17).

4.1 Class Scaling

First, we replicate the experiment of [42, Figure 4] to verify that our model is able to make decisions gradually over longer integration time and can scale to many classes c. Details of the training procedure are relegated to Appendix B.1. For each c, the data distribution is a randomly generated, factorizing distribution on $n = 4$ simplices.

Figure 5 shows the relative entropy between the learned models (histogram of 512k samples) and the known target distribution. Our proposed approach is able to outperform our earlier method [6] (green) as well as Dirichlet flow matching [42] (orange) and the linear flow-matching baseline (blue) in terms of scaling to many classes c. In Fig. 5, the linear flow-matching baseline scales better to many classes than in [42, Figure 4], but the qualitative statement that linear flow matching is ill-suited to this end is still supported by our empirical findings. Our preliminary approach [6] (green) also scales comparatively well, even outperforming Dirichlet flow matching. Figures 9 and 10 illustrate probability paths $\nu_t(\beta)$ for our approach (cf. (3.27)) and Dirichlet flow matching [42] at different time scales.

A property of assignment flow approaches, possibly linked to observed performance, is to transport probability mass relative to the underlying Fisher–Rao geometry (recall Sect. 3.2.5). For example, this leads to little probability mass in regions close to the simplex boundaries (Fig. 9).

4.2 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.

To this end, we parametrize F_θ by the UNet architecture of [16] and train on downsampled segmentations of Cityscapes [12] images, as well as MNIST [35], regarded as binary $c = 2$ segmentations after thresholding continuous pixel values at 0.5. Details of the training procedure are relegated to Appendix B.2.

Figures 6 and 7 show samples from the learned distribution of binarized MNIST and Cityscapes segmentations, respectively, next to the closest training data. This illustrates that our model is able to interpolate the data distribution, without simply memorizing training data. Additional samples from our Cityscapes model are shown in Fig. 8.

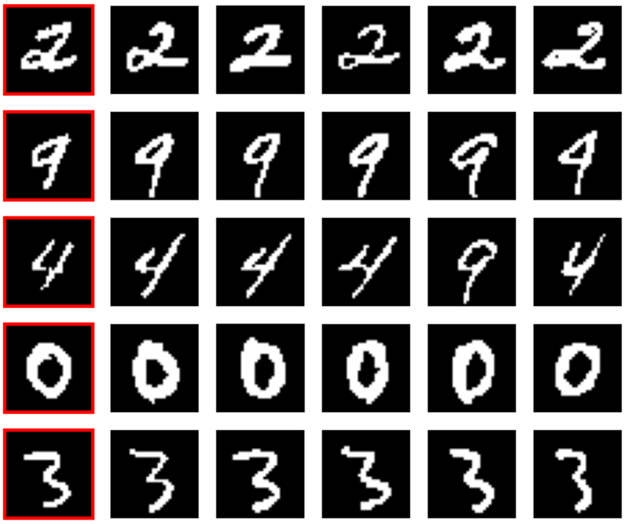


Fig. 6 Comparison of model samples to the closest training data. *Left with red border*: samples drawn from our model of the binarized MNIST distribution. *Right*: training data closest to the sample in terms of pixelwise distance

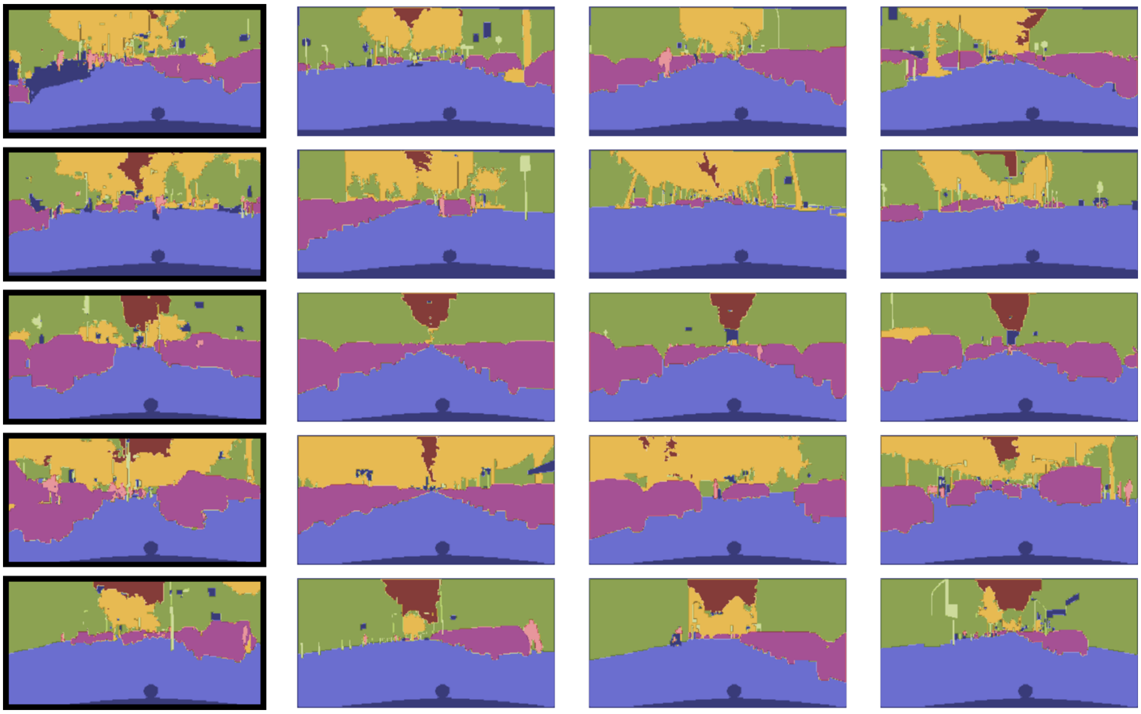


Fig. 7 Comparison of model samples to the closest training data. *Left with black border*: samples drawn from our model of the Cityscapes segmentation distribution. *Right*: training data closest to the sample in terms of pixelwise distance

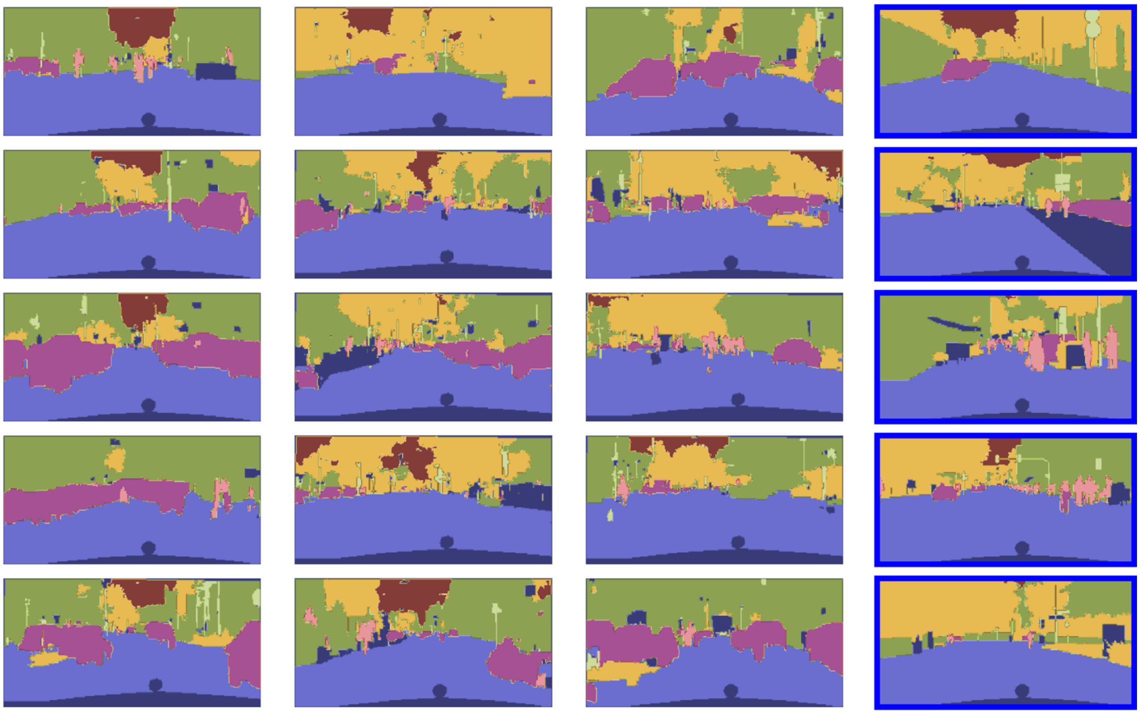


Fig. 8 *Left*: Samples from our model of the Cityscapes segmentation distribution. *Right with blue border*: randomly drawn training data

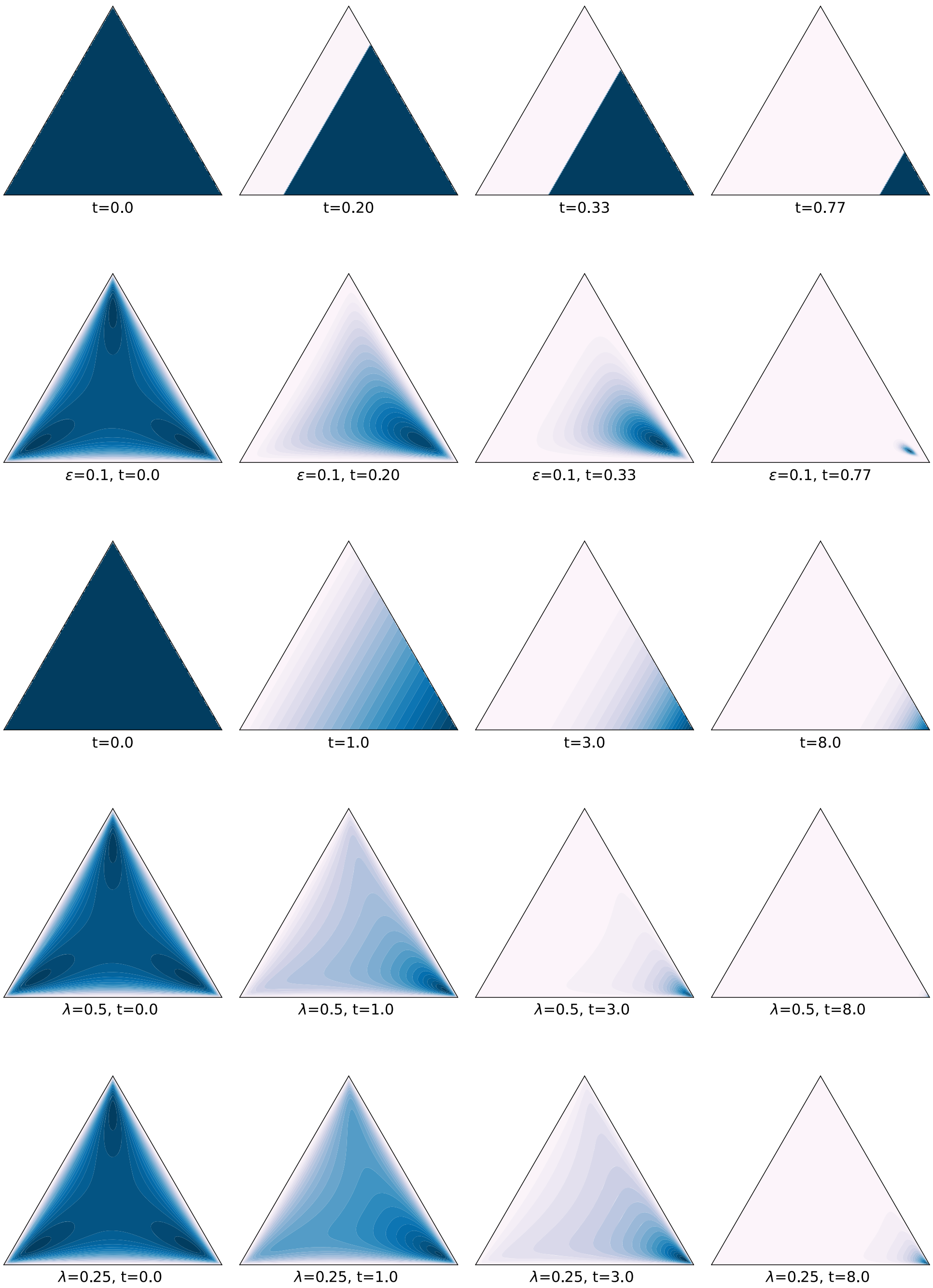


Fig. 9 Plots of conditional densities $\nu_t(\beta)$ for different points of time t. Darker colors indicate higher concentration within the densities. *From top to bottom:* Linear Flow Matching [42, Equation 11], the approach [6, Equation 18], Dirichlet flow matching [42, Equation 14], our approach (3.27) using two different values of the rate parameter λ. Note the different time periods $t \in [0, 0.77]$ used for the first two and $t \in [0, 8]$ for the latter approaches. See Sect. 4.1 for a discussion

4.3 Likelihood Evaluation

We compute the likelihood of test data from the MNIST dataset (binarized by thresholding) using the method described in Sect. 3.5. We use 100 priority samples per datum and, as is common practice for normalizing flows, only a single Hutchinson sample. The result is shown in Table 1, compared to our earlier approach [6] ($t \to 1$). For comparison, we show the likelihood of MNIST test data (from the *continuous*, non-binarized distribution) under several normalizing flow methods from the literature which were trained using likelihood maximization.

Note that, although much prior work on generative modeling has been applied to continuous gray value MNIST images, binarization (in our case through thresholding) substantially changes the data distribution. Thus, likelihood of test data, which is commonly used as a surrogate for relative entropy to the data distribution in normalizing flows, is not comparable between these methods and ours. In addition, since we do not use likelihood maximization as a training criterion, it is not to be expected that our model is compet-

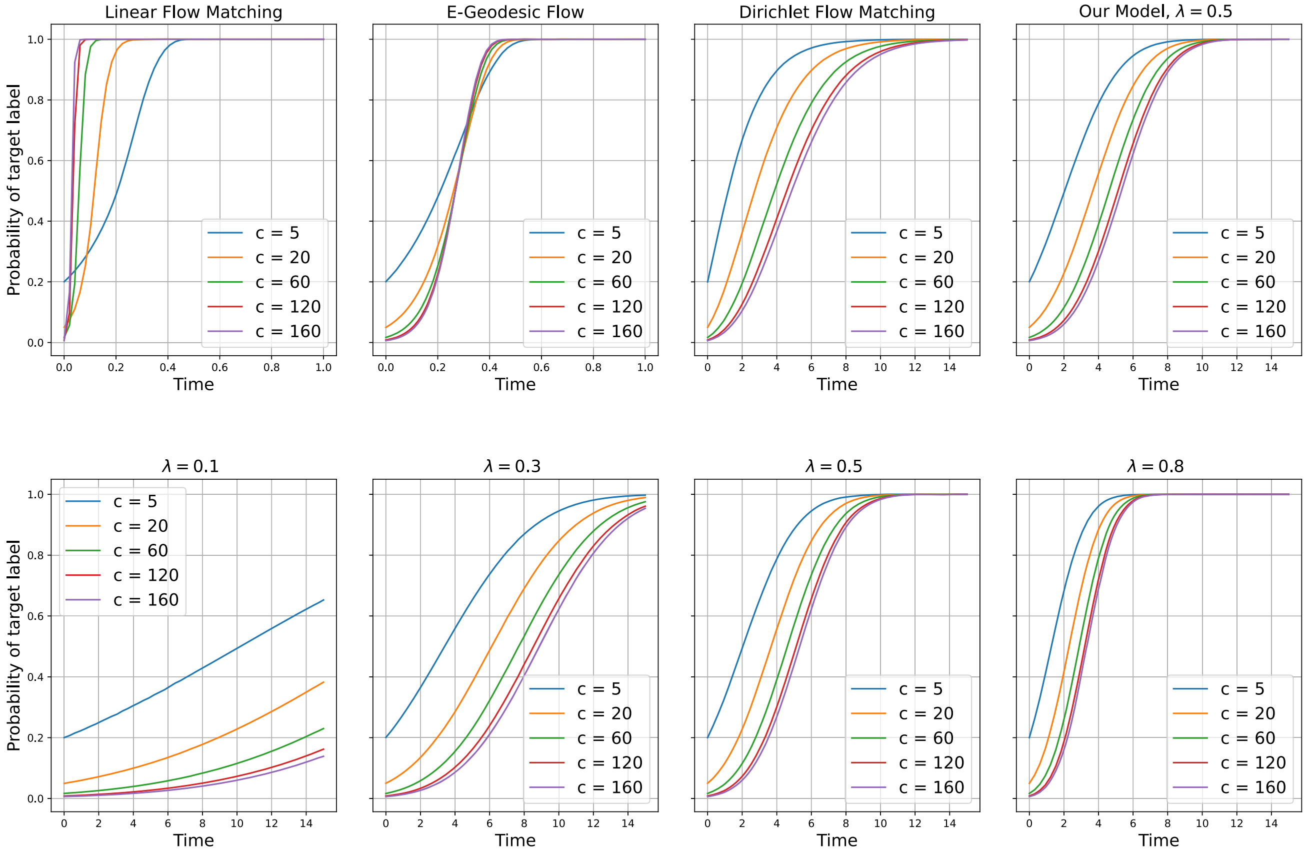


Fig. 10 *Top row:* Plots of conditional densities paths $t \mapsto \nu_t(\beta)$ for various models. *Bottom row:* Impact of the rate parameter λ of our approach (replication of Fig. 3 to ease visual comparison)

itive on this measure. Still, the results of Table 1 indicate that the proposed model ($t \to \infty$) fits the binarized MNIST data distribution better in terms of relative entropy than our previous approach [6] ($t \to 1$).

5 Conclusion

We introduced a novel generative model for the representation and evaluation of joint probability distributions of discrete random variables. The approach employs an embedding of the assignment manifold in the meta-simplex of all joint probability distributions. Corresponding measure transport by randomized assignment flows approximates joint distributions of discrete random variables in a principled manner. The approach enables to learn the statistical dependencies of any set of discrete random variables and using the resulting model for structured prediction, independent of the area of application.

Inference using the approach is computationally efficient, since sampling can be accomplished by parallel geometric numerical integration. Training the generative model using given empirical data is computationally efficient, since matching the flow of corresponding e-geodesics is used as training criterion, which does not require sampling as a subroutine.

Numerical experiments showed superior performance in comparison with the recent related work, which we attribute to consistently using the underlying information geometry of assignment flows and the corresponding measure transport along conditional probability paths. On the other hand, the fact that even our *preliminary* approach [6] can outperform Dirichlet flow matching [42] with respect to scaling to many classes in Fig. 5, is surprising, because the approach [6] uses a *finite* integration time and moves all mass of the reference distribution to a Dirac measure close to $\overline{W}_\beta$ within this finite time. The core assumptions of [42, Proposition 1], therefore, apply to this approach, and the fact that it still performs well empirically suggests that further inquiry into this topic is warranted.

Appendix A. Proofs

A.1 Proofs of Section 3.2.3

Proof of Proposition 3.2 Since V_β is determined by β and does not depend on V, the map $V \mapsto V + \lambda t \lambda V_\beta$ is affine. Hence, Eq. (3.28) conforms to (3.27), because affine transformations of normal distributions are again normal distributions. The mapping $\exp_{\mathbb{1}_{\mathcal{W}}}(\cdot) : \mathcal{W}_c \to \mathcal{T}_0$ is a diffeomorphism. Consequently, the inverse of (3.28) can be computed from

$$W := \psi_t(V|\beta) = \exp_{\mathbb{1}_{\mathcal{W}}}\big(V + t\lambda V_\beta\big) \tag{A.1a}$$

$$\Leftrightarrow \quad \psi_t^{-1}(W|\beta) = V = \exp_{\mathbb{1}_{\mathcal{W}}}^{-1}(W) - t\lambda V_\beta, \tag{A.1b}$$

which verifies (3.29). Regarding (3.30), recall that the conditional flow is determined by the conditional vector field through the ODE

$$\frac{d}{dt}\psi_t(V|\beta) = u_t\big(\psi_t(V|\beta)\big|\beta\big),$$
$$\psi_0(V|\beta) = \psi_0(V) = \exp_{\mathbb{1}_{\mathcal{W}}}(V). \tag{A.2}$$

On the other hand, direct computation of the time derivative of (A.1a) using the closed-form expression

$$d\exp_W(V)[U] = R_{\exp_W(V)}[U] \tag{A.3}$$

for the differential of the lifting map (2.6b), yields

$$\frac{d}{dt}\psi_t(V|\beta) = R_{\psi_t(V|\beta)}[\lambda V_\beta]. \tag{A.4}$$

Equating (A.2) and (A.4) and using $W = \psi_t(V|\beta)$ from (A.1b) proves (3.30). □

Proof of Proposition 3.3 Equation (3.15a) is immediate due to (3.23), (3.24) and (3.27). Writing short

$$\psi_t := \psi_t(\cdot|\beta) \tag{A.5}$$

for the flow map defined by (3.28), it remains to show that

$$\lim_{t\to\infty} \nu_t(\beta) = \lim_{t\to\infty} (\psi_t)_\sharp \nu_0 = \delta_{\overline{W}_\beta}. \tag{A.6}$$

To this end, we demonstrate that every marginal of the conditional probability path (A.6) converges to a Dirac measure supported on the assignment vector corresponding to the labeling configuration β, i.e.,

$$\lim_{t\to\infty} \nu_{t;i}(\beta) = \lim_{t\to\infty} (\psi_{t;i})_\sharp \nu_{0;i} = \delta_{\overline{W}_{\beta;i}}, \quad i \in [n], \tag{A.7}$$

where $\nu_{0;i}$, $i \in [n]$ denote the marginals of ν_0 given by (3.24).

First, we observe that by fixing an orthonormal basis of T_0 as column vectors of the matrix $\mathcal{B}$, every marginal $\nu_{0;i}$ of (3.24) with Gaussian $\mathcal{N}_0$ defined by (3.23) can be expressed as the lifted image measure of a standard normal distribution $\mathcal{N}\big(0_{c-1}, I_{c-1}\big)$ on $\mathbb{R}^{c-1}$ with respect to the basis $\mathcal{B}$,

$$\begin{aligned}\nu_{0;i} &= (\exp_{\mathbb{1}_{\mathcal{S}}})_\sharp \mathcal{B}_\sharp \mathcal{N}\big(\cdot; 0_{c-1}, I_{c-1}\big) \\ &= (\exp_{\mathbb{1}_{\mathcal{S}}})_\sharp \mathcal{N}(\cdot; 0_c, \pi_0),\end{aligned} \tag{A.8}$$

since $\mathcal{B}\mathcal{B}^\top = \pi_0$. Consequently, by Proposition 3.2,

$$\nu_{t;i}(\beta) = \big(\psi_{t;i}\big)_\sharp \mathcal{N}(\cdot; 0_c, \pi_0) \tag{A.9}$$

and hence using the change-of-variables formula and (A.1b), one has for any $p \in \mathcal{S}_c$,

$$\nu_{t;i}(p|\beta) = \mathcal{N}\big(\exp_{\mathbb{1}_{\mathcal{S}}}^{-1}(p) - t\lambda V_{\beta;i}; 0_c, \pi_0\big)|\det d\psi_{t;i}^{-1}|. \tag{A.10}$$

Equation (3.29) shows that the differential $d\psi_{t;i}^{-1}$ does not depend on t. Neither does the normalizing factor of the normal distribution, due to the covariance matrix $\pi_0 = \mathrm{id}_{T_0}$. Consequently, since $\psi_{t;i}^{-1}$ maps to T_0,

$$\nu_t(p|\beta) \propto \exp\Big(-\frac{1}{2}\big\langle \exp_{\mathbb{1}_{\mathcal{S}}}^{-1}(p) - t\lambda V_{\beta;i}, \pi_0\big(\exp_{\mathbb{1}_{\mathcal{S}}}^{-1}(p) - t\lambda V_{\beta;i}\big)\big\rangle\Big) \tag{A.11a}$$

$$= \exp\Big(-\frac{1}{2}\big\langle \exp_{\mathbb{1}_{\mathcal{S}}}^{-1}(p) - t\lambda V_{\beta;i}, \big(\exp_{\mathbb{1}_{\mathcal{S}}}^{-1}(p) - t\lambda V_{\beta;i}\big)\big\rangle\Big) \to 0 \quad \text{as} \quad t \to \infty, \tag{A.11b}$$

for any $p \neq \overline{W}_{\beta;i} \in \overline{\mathcal{S}_c}$ and $i \in [n]$, due to the choice (3.26) of the tangent vector V_β. We conclude that the image measure $\nu_{\infty;i}(\beta)$ is a Dirac measure concentrated on $\overline{W}_{\beta;i}$. □

A.2 Proofs of Section 3.3

Proof of Lemma 3.4 By [8, Lemma 4], one has $Q^\top QV = c^{n-1}V$ for all $V \in \mathcal{T}_0$. Thus, Q_c defined by (3.40) has the property

$${Q_c}^\top Q_c V = V, \qquad \text{for all} \;\; V \in \mathcal{T}_0. \tag{A.12}$$

To show that (3.39) indeed defines the orthogonal projection onto $\operatorname{img} Q \cap \mathcal{T}_0\mathcal{S}_N$, note that

$$Q_c \Pi_0 = \pi_0 Q_c \tag{A.13}$$

by [5, Lemma A.3] and accordingly

$${Q_c}^\top \pi_0 = (\pi_0 Q_c)^\top = (Q_c \Pi_0)^\top = \Pi_0 {Q_c}^\top \tag{A.14}$$

by using the symmetry of Π_0 and π_0. We can use this to show $\operatorname{img}\operatorname{proj}_0 \subseteq \operatorname{img} Q \cap \mathcal{T}_0\mathcal{S}_N$, because for any $x \in \mathbb{R}^{n\times c}$, we have

$$Q_c \Pi_0 x \in \operatorname{img} Q \qquad \text{and} \qquad Q_c \Pi_0 x \overset{(A.13)}{=} \pi_0 Q_c x \in \mathcal{T}_0 \mathcal{S}_N. \tag{A.15}$$

Now let $v \in \mathcal{T}_0\mathcal{S}_N$ and $y \in \operatorname{img} Q \cap \mathcal{T}_0\mathcal{S}_N$ be arbitrary. Then, y can be written as $y = Q_c y'$, and we have

$$\begin{aligned}\langle v - \operatorname{proj}_0(v), y\rangle &= \langle v - Q_c\Pi_0 {Q_c}^\top v, Q_c y'\rangle \\ &= \langle {Q_c}^\top v - {Q_c}^\top Q_c \Pi_0 {Q_c}^\top v, y'\rangle \qquad \text{(A.16a)}\\ &\overset{(A.12)}{=} \langle {Q_c}^\top v - \Pi_0 {Q_c}^\top v, y'\rangle \\ &\overset{(A.14)}{=} \langle {Q_c}^\top v - {Q_c}^\top \pi_0 v, y'\rangle \qquad \text{(A.16b)}\\ &= 0, \qquad \text{(A.16c)}\end{aligned}$$

which shows that proj_0 projects orthogonally. □

Proof of Theorem 3.5 We use the representation of proj_0 (Lemma 3.4) to compute the pushforward (3.42).

$$\begin{aligned}&(\operatorname{proj}_{\mathcal{T}})_\sharp \nu_t^{\mathcal{S}_N}(\beta)\\ &\overset{(3.41)}{=} (\exp_{\mathbb{1}_{\mathcal{S}_N}} \circ \operatorname{proj}_0 \circ \exp^{-1}_{\mathbb{1}_{\mathcal{S}_N}})_\sharp \nu_t^{\mathcal{S}_N}(\beta) \qquad \text{(A.17a)}\\ &\overset{(3.36)}{=} (\exp_{\mathbb{1}_{\mathcal{S}_N}} \circ \operatorname{proj}_0)_\sharp \mathcal{N}_t^{\mathcal{S}_N}(\cdot|\beta) \qquad \text{(A.17b)}\\ &\overset{(3.39)}{=} (\exp_{\mathbb{1}_{\mathcal{S}_N}} \circ Q_c \Pi_0 {Q_c}^\top)_\sharp \mathcal{N}_t^{\mathcal{S}_N}(\cdot|\beta) \qquad \text{(A.17c)}\\ &\overset{(3.35)}{=} (\exp_{\mathbb{1}_{\mathcal{S}_N}} \circ Q_c \Pi_0 {Q_c}^\top)_\sharp \mathcal{N}(\cdot; tc^{n-1}\lambda\pi_0 e_\beta, c^{n-1}\pi_0) \qquad \text{(A.17d)}\\ &= (\exp_{\mathbb{1}_{\mathcal{S}_N}})_\sharp \mathcal{N}\Big(\cdot; tc^{n-1}\lambda Q_c \Pi_0 {Q_c}^\top \pi_0 e_\beta,\\ &c^{n-1} Q_c \Pi_0 {Q_c}^\top \pi_0 (Q_c \Pi_0 {Q_c}^\top)^\top\Big) \qquad \text{(A.17e)}\\ &\overset{\substack{(3.40)\\(A.14)}}{=} (\exp_{\mathbb{1}_{\mathcal{S}_N}})_\sharp \mathcal{N}(\cdot; t\lambda Q \Pi_0 Q^\top e_\beta, c^{n-1} Q_c \Pi_0 {Q_c}^\top Q_c \Pi_0 {Q_c}^\top)\\ &\qquad \text{(A.17f)}\\ &\overset{\substack{(3.40)\\(A.12)}}{=} (\exp_{\mathbb{1}_{\mathcal{S}_N}})_\sharp \mathcal{N}(\cdot; t\lambda Q \Pi_0 Q^\top e_\beta, Q \Pi_0 Q^\top) \qquad \text{(A.17g)}\\ &= (\exp_{\mathbb{1}_{\mathcal{S}_N}} \circ Q)_\sharp \mathcal{N}(\cdot; t\lambda \Pi_0 Q^\top e_\beta, \Pi_0). \qquad \text{(A.17h)}\end{aligned}$$

By [5, Lemma 3.4], we have $Q^\top e_\beta = M e_\beta$, with Q and M defined by (2.9) and (2.14b). Using the shorthand V_β defined by (3.26) and the lifting map lemma (3.38), this shows

$$\begin{aligned}(\operatorname{proj}_{\mathcal{T}})_\sharp \nu_t^{\mathcal{S}_N}(\beta) &= (\exp_{\mathbb{1}_{\mathcal{S}_N}} \circ Q)_\sharp \mathcal{N}(\cdot; t\lambda V_\beta, \Pi_0) \qquad \text{(A.18a)}\\ &\overset{(3.38)}{=} (T \circ \exp_{\mathbb{1}_{\mathcal{W}}})_\sharp \mathcal{N}(\cdot; t\lambda V_\beta, \Pi_0) \qquad \text{(A.18b)}\\ &\overset{(3.26)}{=} (T \circ \exp_{\mathbb{1}_{\mathcal{W}}})_\sharp \mathcal{N}_{t,\beta} \qquad \text{(A.18c)}\\ &\overset{(3.27)}{=} T_\sharp \nu_t(\beta) \qquad \text{(A.18d)}\end{aligned}$$

which is the assertion (3.42).

Returning to (A.18a), we compute the conditional vector field whose flow generates the path $(\operatorname{proj}_{\mathcal{T}})_\sharp \nu_t^{\mathcal{S}_N}(\beta)$ by

$$u_t^{\mathcal{T}}(q|\beta) = d\exp_{\mathbb{1}_{\mathcal{S}_N}}(v)[\lambda Q V_\beta] = R_q[\lambda Q V_\beta] \tag{A.19}$$

with $v = \exp^{-1}_{\mathbb{1}_{\mathcal{S}_N}}(q)$, analogous to (3.30). This shows the shape of the flow-matching criterion (3.43). It remains to show that it is equal to (3.31).

Substituting the ansatz $\widetilde{f}_\theta = Q \circ F_\theta \circ M$ into this criterion gives

$$\begin{aligned}\mathcal{L}^{\mathcal{T}}_{\text{RCFM}} = \mathbb{E}_{t\sim\rho,\beta\sim p,W\sim\nu_t(\beta)}\\ \Big[\big\| R_{T(W)}[\lambda Q(V_\beta) - (Q \circ F_\theta)(W,t)]\big\|^2_{T(W)}\Big].\end{aligned} \tag{A.20}$$

By [5, Theorem 3.1], $T\colon \mathcal{W}_c \to \mathcal{T} \subseteq \mathcal{S}_N$ defined by (2.7) is a Riemannian isometry. Thus, for any vector field $X\colon \mathcal{W}_c \to \mathcal{T}_0$ and any $W \in \mathcal{W}_c$, it holds that

$$\langle R_W[X], R_W[X]\rangle_W = \big\langle dT_W\big[R_W[X]\big], dT_W\big[R_W[X]\big]\big\rangle_{T(W)}. \tag{A.21}$$

Furthermore, by [5, Theorem 3.5], one has

$$dT_W\big[R_W[X]\big] = R_{T(W)}[QX]. \tag{A.22}$$

Taking (A.21) and (A.22) together, (A.20) transforms to

$$\mathcal{L}^{\mathcal{T}}_{\text{RCFM}} = \mathbb{E}_{t\sim\rho,\beta\sim p,W\sim\nu_t(\beta)}\Big[\big\| R_W[\lambda V_\beta - F_\theta(W,t)]\big\|^2_W\Big] \tag{A.23}$$

which is (3.31). □

Appendix B. Experiments: Details

B.1 Details of Class Scaling Experiment

To parameterize F_θ, we use the same convolutional architecture used in [42]. We train for 500 k steps of the Adam optimizer with constant learning rate $3 \cdot 10^{-4}$ and batch size 128. We reproduce the Dirichlet flow-matching results and linear flow-matching baseline by using the code of [42]. The experiment shown in Fig. 5 is slightly harder than the version in [42], because we limit training to 64 k steps at batch size 512 for Dirichlet- and linear flow matching. Accordingly, both assignment flow methods are trained for 250 k steps at batch size 128, such that around 32M data are seen by each model during training.

B.2 Details of Generating Image Segmentations

B.2.1 Cityscapes Data Preparation

Rather than the original $c = 33$ classes, we only use the $c = 8$ classes specified as *categories* in *torchvision*. The same subsampling of classes was used in the related work [26]. They additionally perform spatial subsampling to 32×64. Instead, we subsample the spatial dimensions (*NEAREST* interpolation) to 128×256.

B.2.2 Cityscapes Training

For the Cityscapes experiment, we employ the UNet architecture of [16] 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 cosine annealing learning rate scheduler starting at learning rate 0.0003 and batch size 4. The distribution ρ of times t used during training is an exponential distribution with rate parameter $\lambda = 0.25$. For sampling, we integrate up to $t_{\max} = 15$.

B.2.3 Binarized MNIST Data Preparation

We pad the original 28×28 images with zeros to size 32×32 to be compatible with spatial downsampling employed by the UNet architecture. Binarization is performed by pixelwise thresholding at gray value of 0.5.

B.2.4 Binarized MNIST Training

We modify the same architecture used for Cityscapes to *attention_resolutions* (16), *channel_mult* (1,2,2,2), 4 attention heads, 2 blocks and 32 channels. The same training regimen is used as for Cityscapes except for an increase in batch size to 256. The distribution ρ of times t used during training is an exponential distribution with rate parameter $\lambda = 0.5$. For sampling, we integrate up to $t_{\max} = 10$. In Table 1, we use the same UNet architecture and training regimen for the comparison method [6] ($t \to 1$).

Appendix C. Likelihood Computation: Details

Assume we have learned a probability path ν_t and a final pushfoward distribution ν_∞. In practice, numerical integration needs to be stopped after a finite time $t = t_{\max}$, reaching a numerical pushforward distribution $\nu_{t_{\max}} \approx \nu_\infty$. Drawing samples from $\widetilde{p} = \mathbb{E}_{W\sim\nu_{t_{\max}}}[T(W)]$ is a two-stage process: $W \sim \nu_{t_{\max}}$ is drawn first, followed by sampling $\beta \sim T(W)$. Due to the numerical need to stop integration at finite time, $T(W)$ may in practice not have fully reached a discrete Dirac distribution. For long sequences of random variables, such as text or image modalities, this can lead to undesirable noise in the output samples. A way to combat this numerical problem is by rounding to a Dirac measure before sampling. This procedure can be interpreted within the framework of *dequantization*, which we elaborate in Sect. 3.6.

In practice, $W \sim \nu_{t_{\max}}$ is typically close to a discrete Dirac already (cf. Fig. 4), so rounding has little impact on the represented joint distribution. Nevertheless, the rounding process is formally a different model than $\widetilde{p} = \mathbb{E}_{W\sim\nu_{t_{\max}}}[T(W)]$, which we explicitly distinguish for the purpose of computing likelihoods. Recall the definition (3.32) of subsets $r_\beta \subseteq \mathcal{W}_c$ with each $W \in r_\beta$ assigning the largest probability to the labels β. The points in r_β are also the ones which round to $\overline{W}_\beta$.[2] Thus, the labeling $\beta \in [c]^n$ has likelihood

$$\widetilde{p}^r_\beta = \mathbb{E}_{W\sim\nu_{t_{\max}}}[1_{r_\beta}(W)] = \mathbb{P}_{\nu_{t_{\max}}}(r_\beta) \tag{C.1}$$

under the rounding model $\widetilde{p}^r$, with 1_{r_β} denoting the indicator function of r_β. This is numerically similar to the likelihood under our original model

$$\widetilde{p}_\beta = \mathbb{E}_{W\sim\nu_\infty}[T(W)_\beta] \tag{C.2}$$

and matches it in the limit $t \to \infty$, provided that (almost) every trajectory $W(t)$ approaches an extreme point of $\overline{\mathcal{W}_c}$ under the learned assignment flow dynamics.

We will now devise an importance sampling scheme for efficient and numerically stable approximation of the integral in (C.1), that analogously applies to (C.2). Let ϱ be a proposal distribution with full support on $\mathcal{W}_c$ which has most of its mass concentrated around a point $q_\beta \in \mathcal{W}_c$ close to $\overline{W}_\beta$. Then

$$\mathbb{P}_{\nu_{t_{\max}}}(r_\beta) = \mathbb{E}_{W\sim\varrho}\Big[1_{r_\beta}(W)\frac{\nu_{t_{\max}}(W)}{\varrho(W)}\Big] \tag{C.3}$$

where we assumed that both $\nu_{t_{\max}}$ and ϱ have densities with respect to the Lebesgue measure and used again the symbols $\nu_{t_{\max}}$ and ϱ to denote these densities. The rationale behind this construction is that, since we learned $\nu_{t_{\max}}$ to concentrate close to points $\overline{W}_\beta$, drawing most samples close to q_β will reduce the estimator variance compared to sampling (C.1) directly. In high dimensions, the quantities in (C.3) are prone to numerical underflow, which motivates the transformation

$$\begin{aligned}\log \mathbb{P}_{\nu_{t_{\max}}}(r_\beta) &= \log \mathbb{E}_{W\sim\varrho}\Big[1_{r_\beta}(W)\frac{\nu_{t_{\max}}(W)}{\varrho(W)}\Big] &\text{(C.4a)}\\ &= \log \mathbb{E}_{W\sim\varrho}\big[\exp\big(\log 1_{r_\beta}(W) \\ &\qquad + \log \nu_{t_{\max}}(W) - \log\varrho(W)\big)\big]. &\text{(C.4b)}\end{aligned}$$

After replacing the expectation with a mean over samples drawn from ϱ, we can evaluate (C.4) by leveraging stable numerical implementations of the logsumexp function.

[2] The sets r_β technically overlap on the boundary, but all intersections have measure zero.

For every evaluation of the integrand, we evaluate log-likelihood under ϱ in a closed form as well as log-likelihood under $\nu_{t_{\max}}$ through numerical integration backward in time, leveraging the instantaneous change of variables (3.45) and Hutchinson's trace estimator (3.46). Note the conventions $\log 0 = -\infty$ and $\exp(-\infty) = 0$ employed in (C.4). The analogous expression for (C.1) reads

$$\log \widetilde{p}_\beta = \log \mathbb{E}_{W\sim\varrho}\big[\exp\big(\log T(W)_\beta + \log \nu_{t_{\max}}(W) - \log \varrho(W)\big)\big] \tag{C.5}$$

and we can further expand

$$\log T(W)_\beta = \log \prod_{i\in[n]} W_{i,\beta_i} = \sum_{i\in[n]} \log W_{i,\beta_i} \tag{C.6}$$

to avoid numerical underflow.

Acknowledgements This work was funded by the Deutsche Forschungsgemeinschaft (DFG), Grant SCHN 457/17-2, within the priority programme SPP 2298: Theoretical Foundations of Deep Learning. This work was funded by the Deutsche Forschungsgemeinschaft (DFG) under Germany's Excellence Strategy EXC-2181/1-390900948 (the Heidelberg STRUCTURES Excellence Cluster).

Author Contributions Each author was involved in the drafting of the manuscript and revising it. All authors reviewed and approved the final version of the manuscript.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability No datasets were generated or analyzed during the current study.

Declarations

Conflict of interest The authors declare no conflict of interest.

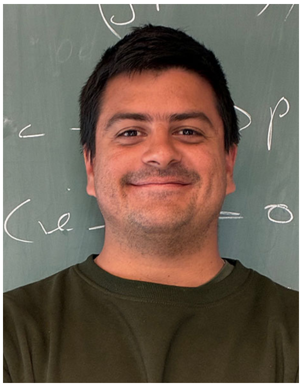
References

1. Agresti, A.: Categorial Data Analysis, 3rd edn. Wiley, New York (2013)
2. Aitchinson, J.: The statistical analysis of compositional data. J. R. Stat. Soc. B **2**, 139–177 (1982)
3. Amari, S.-I., Nagaoka, H.: Methods of Information Geometry. American Mathematical Society, Providence (2000)
4. Åström, F., Petra, S., Schmitzer, B., Schnörr, C.: Image labeling by assignment. J. Math. Imaging Vis. **58**(2), 211–238 (2017)
5. Boll, B., Cassel, J., Albers, P., Petra, S., Schnörr, C.: A Geometric Embedding Approach to Multiple Games and Multiple Populations, preprint arXiv:2401.05918 (2024)
6. Boll, B., Gonzalez-Alvarado, D., Schnörr, C.: Generative Modeling of Discrete Joint Distributions by E-Geodesic Flow Matching on Assignment Manifolds, preprint arXiv:2402.07846 (2024)
7. Boll, B., Schwarz, J., Gonzalez-Alvarado, D., Sitenko, D., Petra, S., Schnörr, C.: Modeling large-scale joint distributions and inference by randomized assignment, scale space and variational methods in computer vision (SSVM). In: Calatroni, L., Donatelli, M., Morigi, S., Prato, M., Santacesaria, M. (eds.), LNCS, no. 14009. Springer, pp. 730–742 (2023)
8. Boll, B., Schwarz, J., Schnörr, C.: On the Correspondence between Replicator Dynamics and Assignment Flows, SSVM 2021: Scale Space and Variational Methods in Computer Vision, LNCS, vol. 12679. Springer, pp. 373–384 (2021)
9. Chen, R.T.Q., Amos, B., Nickel, M.: Semi-Discrete Normalizing Flows through Differentiable Tesselation, NeurIPS (2022)
10. Cowell, R.G., Dawid, A.P., Lauritzen, S.L., Spiegelhalter, D.J.: Probabilistic Networks and Expert Systems. Springer, Berlin (1999)
11. Chen, R.T.Q., Lipman, Y.: Riemannian Flow Matching on General Geometries, preprint arXiv:2302.03660 (2023)
12. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
13. Chen, R.T.Q., Rubanova, Y., Bettencourt, J., Duvenaud, D.: Neural ordinary differential equations. In: Proceedings of the NeurIPS (2018)
14. Cover, T.M., Thomas, J.A.: Elements of Information Theory, 2nd edn. Wiley, New York (2006)
15. Davis, O., Kessler, S., Petrache, M., Ceylan, I.I., Bronstein, M., Bose, A.J.: Fisher Flow Matching for Generative Modeling over Discrete Data, preprint arXiv:2405.14554 (2024)
16. Dhariwal, P., Nichol, A.: Diffusion models beat GANs on image synthesis. In: NeurIPS (2021)
17. Dormand, J.R., Prince, P.J.: A family of embedded Runge–Kutta Formulae. J. Comput. Appl. Math. **6**(1), 19–26 (1980)
18. Dinh, L., Sohl-Dickstein, J., Bengio, S.: Density estimation using real NVP. In: ICLR (2017)
19. Drton, M., Sturmfels, B., Sullivant, S.: Lecture on Algebraic Statistics, Oberwolfach Seminars, vol. 39, Birkhäuser (2009)
20. Ferguson, T.S.: A Bayesian analysis of some nonparametric problems. Ann. Stat. **2**(1), 209–230 (1973)
21. Grathwohl, W., Chen, R.T.Q., Bettencourt, J., Sutskever, I., Duvenaud, D.: FFJORD: free-form continuous dynamics for scalable reversible generative models. In: ICLR (2019)
22. Geiger, D., Meek, C., Sturmfels, B.: On the Toric algebra of graphical models. Ann. Stat. **34**(3), 1463–1492 (2006)
23. Harris, J.: Algebraic Geometry: A First Course. Springer, Berlin (1992)
24. Ho, J., Chen, X., Srinivas, A., Duan, Y., Abbeel, P.: Flow++: improving flow-based generative models with variational dequantization and architecture design. In: Proceedings of the ICML, vol. PMLR 97, pp. 2722–2730 (2019)
25. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration. Springer, Berlin (2006)
26. Hoogeboom, E., Nielsen, D., Jaini, P., Forré, P., Welling, M.: Argmax flows and multinomial diffusion: learning categorial distributions. In: NeurIPS (2021)

27. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I, 3rd edn. Springer, Berlin (2008)
28. Hofbauer, J., Sigmund, K.: Evolutionary Games and Population Dynamics. Cambridge University Press, London (1998)
29. Hutchinson, M.F.: A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. Commun. Stat. Simul. Comput. **18**(3), 1059–1076 (1989)
30. Johnson, N.L., Kotz, S.: Urn Models and Their Application. Wiley, New York (1977)
31. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press, London (2009)
32. Kobyzev, I., Prince, S.J.D., Brubaker, M.A.: Normalizing flows: an introduction and review of current methods. IEEE Trans. Pattern Anal. Mach. Intell. **43**(11), 3964–3979 (2021)
33. Landsberg, J.M.: Tensors: Geometry and Applications. American Mathematical Society, Providence (2012)
34. Lauritzen, S.L.: Graphical Models. Clarendon Press, Oxford (1996)
35. LeCun, Y., Cortes, C., Burges, C.J.: MNIST Handwritten Digit Database, vol. 2. ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist (2010)
36. Lipman, Y., Chen, R.T.Q., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: ICLR (2023)
37. Li, W., Montufar, G.: Natural gradient via optimal transport. Inf. Geom. **2**(1), 181–214 (2018)
38. Lin, S., Sturmfels, B., Xu, Z.: Marginal likelihood integrals for mixtures of independence models. J. Mach. Learn. Res. **10**, 1611–1631 (2009)
39. Munthe-Kaas, H.: High order Runge–Kutta methods on manifolds. Appl. Numer. Math. **29**(1), 115–127 (1999)
40. Papamakarios, G., Nalisnick, E., Rezende, D.J., Mohamed, S., Lakshminarayanan, B.: Normalizing flows for probabilistic modeling and inference. J. Mach. Learn. Res. **22**(57), 1–64 (2021)
41. Ruthotto, L., Haber, E.: An introduction to deep generative modeling. GAMM Mitt. **44**(2), e202100008 (2021)
42. Stark, H., Jing, B., Wang, C., Corso, G., Berger, B., Barzilay, R., Jaakkola, T.: Dirichlet Flow Matching with Applications to DNA Sequence Design, preprint arXiv:2402.05841 (2024)
43. Salimans, T., Karpathy, A., Chen, X., Kingma, D.P.: PixelCNN++: improving the pixelCNN with discretized logistic mixture likelihood and other modifications. In: ICLR (2017)
44. Studený, M.: On Probabilistic Conditional Independence Structures. Springer, Berlin (2005)
45. Sullivant, S.: Algebraic Statistics. American Mathematical Society, Providence (2018)
46. Theis, L., van den Oord, A., Bethge, M.: A note on the evaluation of generative models. In: ICLR (2016)
47. Uria, B., Murray, I., Larochelle, H.: RNADE: the real-valued neural autoregressive density-estimator. In: NIPS (2013)
48. Zeilmann, A., Savarino, F., Petra, S., Schnörr, C.: Geometric numerical integration of the assignment flow. Inverse Prob. **36**(3), 034004 (2020)
49. Zwiernik, P.: Semialgebraic Statistics and Latent Tree Models. CRC Press, Boca Raton (2016)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Bastian Boll received his M.Sc. degree in Scientific Computing (2019) and his Ph.D. degree in Mathematics (2024) from Heidelberg University, Germany, where he continued his work as a postdoctoral researcher in the Image and Pattern Analysis group of Christoph Schnörr until February 2025. He is now an AI Researcher at Aleph Alpha GmbH. His research interests include geometric and statistical methods for structured prediction, PAC-Bayesian learning theory, and their application to discrete data such as graphs, text, and 3D segmentation.



Daniel Gonzalez-Alvarado received his M.Sc. degree in Mathematics (2020) from Heidelberg University, Germany. Currently, he is a Ph.D. student in the Image and Pattern Analysis group led by Christoph Schnörr. His research interests include generative modeling, uncertainty quantification, machine learning, and applications in image analysis.

Stefania Petra received her B.Sc. in Mathematics and Computer Science (2001) and M.Sc. in Mathematics (2003) from Babe s-Bolyai University of Cluj-Napoca. She obtained her Ph.D. in numerical optimization from the University of Würzburg in 2006. Afterward, she worked as a research fellow at the Universities of Mannheim and Heidelberg, focusing on mathematical image processing. From 2013 to 2015, she was a Margarete von Wrangel Fellow in the postdoctoral qualification program of the Ministry of Science, Research, and Arts of Baden-Württemberg. Between 2015 and 2023, she served as an Assistant Professor at Heidelberg University. Since 2023, Stefania Petra has been a Professor at Augsburg University, leading the chair on Mathematical Imaging at the Institute of Mathematics and the Centre for Advanced Analytics and Predictive Sciences (CAAPS). Her research interests include mathematical models of image analysis, particularly in numerical optimization and information geometry.

Christoph Schnörr received his degrees from the Technical University of Karlsruhe (1991) and the University of Hamburg (1996), respectively. He became full professor at the University of Mannhe im in 1998. In 2008 he joined the Heidelberg University where he is heading the Image and Pattern Analysis Group at the Institute for Mathematics. He is member of the Research Station Geometry + Dynamics, associate member of the Interdisciplinary Center for Scientific Computing (IWR) and steering board member of the cluster of excellence STRUCTURES. He was co-Editor in Chief of the International Journal of Computer Vision (2005-2014) and is member of the Editorial Boards of the SIAM Journal of Imaging Sciences (2016-2025) and the Journal of Mathematical Imaging and Vision. His research focuses on mathematical image and data analysis with a focus on variational methods, information and differential geometry, dynamical systems and machine learning.