Introduction to Geometric Deep Learning

(or Deep Learning Geometrically)

Leo Zhang

University of Oxford

Feburary 10, 2026

Research Interests

Third-year DPhil student supervised by Rob Cornish, Saifuddin Syed and Yee Whye Teh.

1. Generative Modelling:
   - Diffusion models etc.
2. Sampling:
   - MCMC, Neural samplers, Parallel Tempering, reward tilting
3. Molecular Generation:
   - De novo molecular generation, protein-ligand docking, co-folding
4. Geometry/Inductive Biases/Equivariance:
   - $\mathrm{SE}(3)$ Diffusion, Stochastic Equivariance
5. Categorical Probability/Architecture Design:
   - How can we think about neural networks via category theory?

Contents

Introduction

No Free Lunch

Deep learning has been extraordinarily successful in learning from natural data—i.e. images, language, molecules.

- One explanation is the *Universal Approximation Theorem*.
- However, "no free lunch" results imply no single "optimal" learning algorithm exists for all possible distributions.

The Structure of Data

We do not care about the entire space of distributions such as fitting random noise.

- Natural data is structured, notably in terms of its *geometry*.
- This is the focus of this talk.

Other notions of structure that we do not focus on include:

- Simplicity bias from physical constraints/Kolmogorov complexity of natural data.
- Compositionality and hierarchy of features.

Overview

We will explore how the geometric structure of data (and neural networks) can be understood and exploited in deep learning.

The Geometry of Data

The main notions of geometric structure arise in terms of the manifold hypothesis, data living on a manifold and symmetry:

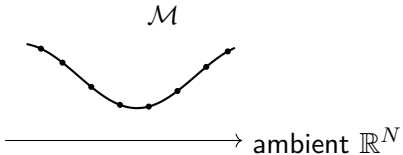
- Natural data is **high-dimensional** but tends to exhibit **low-dimensional structure**.
- Some data can also be considered as existing on a **manifold**.
- Many learning problems have **symmetries**: transformations that should not change the answer.

Note

Geometry provides a mathematical language to encode both as **manifolds** and **groups**.

Manifold Hypothesis

- Natural data often varies along a few underlying factors (pose, lighting, style etc.) leading to low-dimensional structure.
- ML Methods:
  - Dimensionality reduction (e.g. PCA, t-SNE),
  - Topological data analysis
  - Manifold/Metric learning

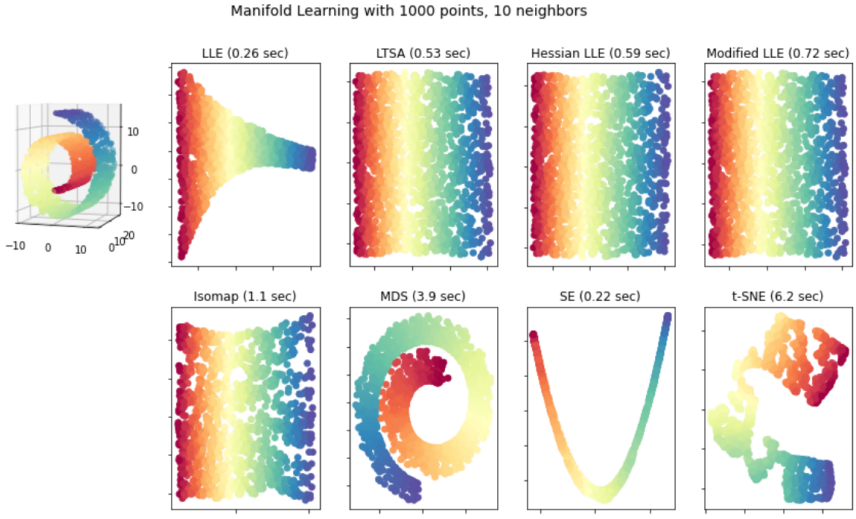


Definition (Manifold hypothesis (informal))

There exists a low-dimensional manifold $\mathcal{M} \subset \mathbb{R}^N$ with $\dim(\mathcal{M}) = n \ll N$ such that most data lie near $\mathcal{M}$:

$$\mathbb{P}_{\text{data}}(\{x : \operatorname{dist}(x, \mathcal{M}) \leq \varepsilon\}) \approx 1 \quad (\varepsilon \text{ small}).$$

Manifold Learning



Applications to ML Theory

Linear Convergence of Diffusion Models Under the Manifold Hypothesis

Peter Potaptchik*, Iskander Azangulov*, and George Deligiannidis

University of Oxford
{surname}@stats.ox.ac.uk

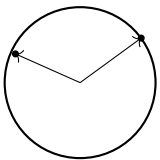
Abstract

Score-matching generative models have proven successful at sampling from complex high-dimensional data distributions. In many applications, this distribution is believed to concentrate on a much lower d-dimensional manifold embedded into D-dimensional space; this is known as the manifold hypothesis. The current best-known convergence guarantees are either linear in D or polynomial (superlinear) in d. The latter exploits a novel integration scheme for the backward SDE. We take the best of both worlds and show that the number of steps diffusion models require in order to converge in Kullback-Leibler (KL) divergence is linear (up to logarithmic terms) in the intrinsic dimension d. Moreover, we show that this linear dependency is sharp.

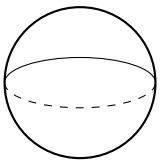
Data Living on Manifolds

- Data can be **intrinsically constrained**:
  - angles on $\mathbb{S}^1$, directions on $\mathbb{S}^2$,
  - rotations on $\mathrm{SO}(3)$, rigid poses on $\mathrm{SE}(3)$,
  - covariance matrices (SPD manifold), probability simplices.
- Data **parametrised** in terms of manifolds can also provide a useful inductive bias.
- Why not just embed them in Euclidean space?
  - Euclidean distances/losses distort the intrinsic notion of "closeness".
  - E.g. coordinate singularities (Euler angles) and wrap-around,
- Geometry provides **intrinsic operations**: geodesic distance, exponential/log maps, parallel transport.

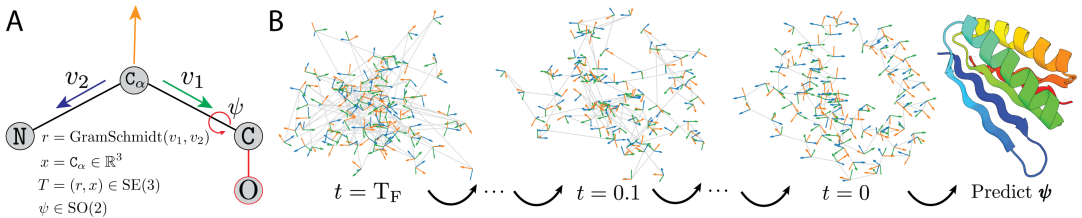
$\mathbb{S}^1$ (angles)



$\mathbb{S}^2$ (directions)

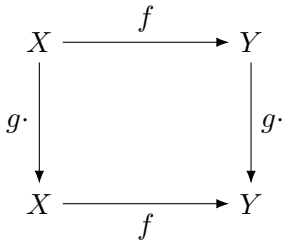


SE(3) Diffusion Models for Proteins



Symmetries

- A **symmetry** is a transformation of the input that should induce a predictable transformation of the output.
- Examples:
  - Images: translations/rotations/reflections.
  - Graphs: node permutations.
  - Molecules/proteins: rigid motions $\mathrm{SE}(3)$.
  - Climate and astrophysics: rotations on the sphere.



Group Theory

Definition (Group)

A **group** $(G, \cdot)$ is a set with an associative product, an identity element e, and inverses: for every $g \in G$ there exists $g^{-1} \in G$.

Definition (Group action)

A (left) action of G on a set X is a map $(g, x) \mapsto g \cdot x$ such that:

- $e \cdot x = x$
- $(gh) \cdot x = g \cdot (h \cdot x)$

Core examples

- Translations: $(\mathbb{R}^d, +)$.
- Rotations: $\mathrm{SO}(2), \mathrm{SO}(3)$.
- Rigid motions: $\mathrm{SE}(3) = \mathbb{R}^3 \rtimes \mathrm{SO}(3)$.
- Permutations: S_n acts on graphs.

Why groups?

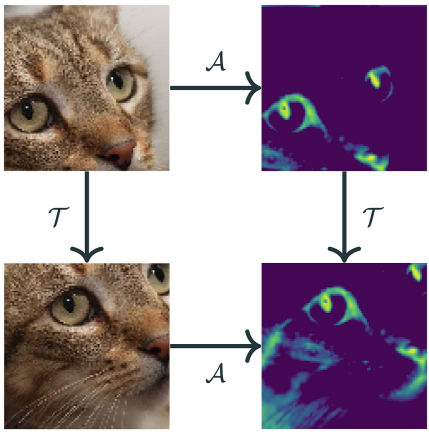
They provide the mathematical formalising for capturing symmetries.

Equivariance

Invariance vs equivariance

For a group action $g \cdot x$ on inputs and $g \cdot y$ on outputs and a function $f : X \to Y$:

- **Equivariance:** $f(g \cdot x) = g \cdot f(x)$
- **Invariance:** $f(g \cdot x) = f(x)$



Invariance vs Equivariance on the Sphere

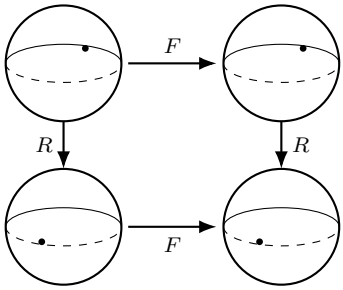
- Let $x : \mathbb{S}^2 \to \mathbb{R}$ be a global field (temperature, pressure, ...).
- A **scalar prediction** ("is there a storm?") should be **rotation-invariant**:

$$f(R \cdot x) = f(x).$$

- A **map prediction** ("wind fields") should be **rotation-equivariant**:

$$F(R \cdot x) = R \cdot F(x).$$

- Equivariance is a strong prior: it holds for *all* rotations, including unseen ones.



Geometric Deep Learning

The guiding aim of *Geometric Deep Learning* (Bronstein et al., 2021) is the design of neural network architectures which incorporate the underlying symmetries of the data domain through equivariance.

Domains

- Grids (images)
- Graphs
- Point clouds
- Manifolds

Aims

- Data efficiency
- Faster learning
- Robustness/Generalisation
- Physically meaningful models

Unifying viewpoint

CNNs are translation-equivariant; Transformers/GNNs are permutation-equivariant; GDL generalises this to arbitrary groups and manifolds.

Geometric Deep Learning

Approaches for constructing equivariance can be roughly categorised into:

- **Intrinsic**: each layer is constructed to be equivariant (e.g. group convolution)
  - This approach is often quite involved and tends to suffer from increased computational cost and optimisation difficulties
- **Extrinsic**: the entire model is converted to be equivariant (e.g. symmetrisation)
  - This approach can utilise standard architectures but symmetrisation suffers from needing multiple forward passes and Monte Carlo error

Stochastic Equivariance

See Bloem-Reddy and Teh (2020); Cornish (2024); Zhang et al. (2024)

SymDiff: Equivariant Diffusion via Stochastic Symmetrisation

Leo Zhang **Kianoosh Ashouritaklimi** **Yee Whye Teh** **Rob Cornish**
Department of Statistics, University of Oxford

ABSTRACT

We propose SymDiff, a method for constructing equivariant diffusion models using the framework of stochastic symmetrisation. SymDiff resembles a learned data augmentation that is deployed at sampling time, and is lightweight, computationally efficient, and easy to implement on top of arbitrary off-the-shelf models. In contrast to previous work, SymDiff typically does not require any neural network components that are intrinsically equivariant, avoiding the need for complex parameterisations or the use of higher-order geometric features. Instead, our method can leverage highly scalable modern architectures as drop-in replacements for these more constrained alternatives. We show that this additional flexibility yields significant empirical benefit for $\mathrm{E}(3)$-equivariant molecular generation. To the best of our knowledge, this is the first application of symmetrisation to generative modelling, suggesting its potential in this domain more generally.

Deep Learning Geometrically

The geometric considerations can also be useful to deep learning more broadly than just handling the geometry of data. For instance,

- Controlling signal propagation via neural network initialisation (Xiao et al., 2018)
- Handling vanishing gradients in RNNs (Helfrich et al., 2018)
- Avoiding rank collapse in Transformers (Zhang and Martens, 2026)
- Optimisers (K-FAC, Muon etc.)
- Understanding contrastive learning (Wang and Isola, 2020)

Orthogonal Initialisation for Dynamical Isometry

From Xiao et al. (2018):

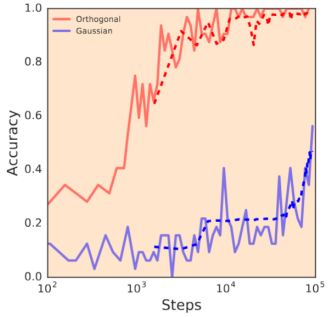


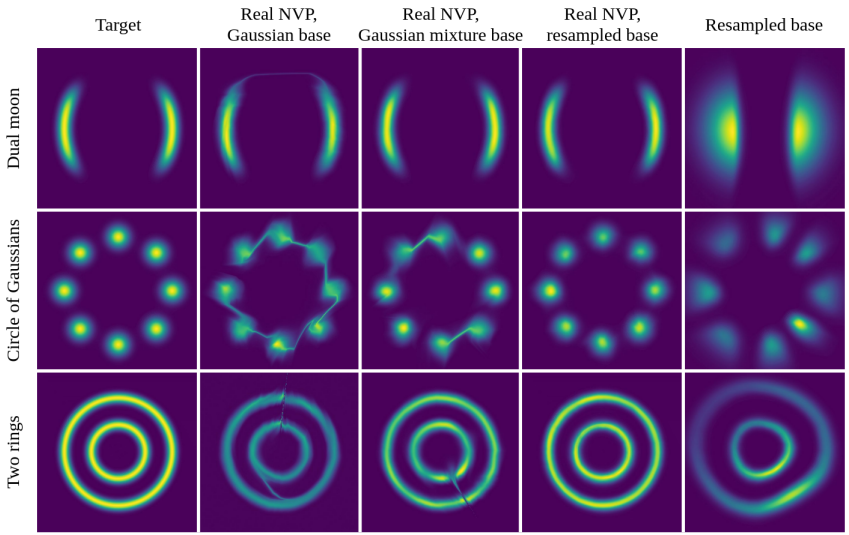
Figure 5. Orthogonal initialization leads to faster training in CNNs. Training (solid lines) and test curves for a 4,000-layer CNN trained using orthogonal (red) and Gaussian (blue) initializations with identical weight variance.

Topology Mismatch in Normalising Flows

As an example of how geometric reasoning can be used for reasoning about the design and behaviour of deep learning models, consider normalising flows.

- Normalising flows utilise a diffeomorphism to map from a noise distribution to data.
- The noise distribution is usually taken to be a Gaussian for simplicity.
- However, this creates a topological mismatch when the data distribution is multimodal.
- Cornish et al. (2020) proves that this results in expressivity and optimisation limitations.

Topology Mismatch in Normalising Flows



Mathematical Requirements

- **Group theory**: groups and group actions.
- **Representation theory**: how to construct equivariant layers.
- **Riemannian geometry**: calculus and optimisation on curved spaces.
- **Lie groups/algebras**: $\mathrm{SO}(3)$, $\mathrm{SE}(3)$, matrix exponentials.
- **Probability on manifolds**: densities, SDEs, Brownian motion.

Topics

- Manifold-Valued Generative Models
- Architecture Design
- Equivariance?
- Optimisation

Other Topics

- Natural gradient optimisers
- Otto calculus
- Information geometry
- Optimal transport
- ... and many more

Manifold-Valued Generative Models

Generative Models

- Goal: learn a distribution $p_\theta(x)$ from which we can sample that matches $p_{\text{data}}(x)$.
- Common approaches:
  - Normalising flows (Mathieu and Nickel, 2020),
  - Diffusion / Flow models (De Bortoli et al., 2022; Chen and Lipman, 2023).
  - Specialise to specific manifolds such as $\mathrm{SE}(3)$ (Yim et al., 2023).

Score-based diffusion models in $\mathbb{R}^n$

Define a SDE which transports p_{data} to a Gaussian. We can then sample from p_{data} from the reverse-time SDE which depends on the score $\nabla_x \log p_t(x)$. We can learn this quantity with a neural network via the score matching objective.

What goes wrong on manifolds?

How do we define SDEs on manifolds intrinsically? Do we have an equivalent objective for manifolds?

Riemannian Geometry: a Basic Toolkit

- **Tangent spaces** T_pM: local linearisation.
- **Riemannian metric** g: turns tangent vectors into lengths/angles.
- **Geodesics**: shortest paths; define intrinsic interpolation.
- **Exponential / logarithm maps**: move between T_pM and M.
- **Volume element** dVol_g: integration, probability densities.

Why we care in generative modelling

We need stochastic processes, gradients, change-of-variables formulas, and dynamics that respect the geometry.

I would recommend Lee (2003) for learning more.

Smooth Manifolds

Definition

A topological space M is a smooth (n-dimensional) manifold if

- M is Hausdorff and second-countable;
- M has an atlas of coordinate charts $\{(U, \varphi)\}$ where $U \subset M$ is open, $\varphi : U \to \tilde{U} \subset \mathbb{R}^n$ is a homeomorphism, and the $\{U\}$ cover M;
- for all charts $(U, \varphi), (V, \psi)$ with $U \cap V \neq \emptyset$, the transition map

$$\psi \circ \varphi^{-1} : \varphi(U \cap V) \to \psi(U \cap V)$$

  is a diffeomorphism.

Remark

Charts let us define calculus on M while keeping constructions coordinate-free.

Smooth Manifolds

Coordinate charts provide local coordinates

$$\varphi(p) = \big(x^1(p), \dots, x^n(p)\big) \in \mathbb{R}^n, \qquad p \in U.$$

Definition (Smooth maps)

Let M, N be smooth manifolds and $F : M \to N$. We say F is smooth at $p \in M$ if there exist charts (U, φ) around p and (V, ψ) around $F(p)$ such that $F(U) \subset V$ and the Euclidean map

$$\psi \circ F \circ \varphi^{-1} : \varphi(U) \to \psi(V)$$

is smooth.

Smooth Manifolds: Examples

Example: Euclidean space

$\mathbb{R}^n$ is a smooth manifold with a single global chart (the identity map).

Example: the sphere $\mathbb{S}^2$

$\mathbb{S}^2 = \{x \in \mathbb{R}^3 : \|x\| = 1\}$ is a 2D manifold. A standard atlas uses stereographic projections.

Let $N = (0, 0, 1)$ and $S = (0, 0, -1)$. Define charts

$$\sigma_N : \mathbb{S}^2 \setminus \{N\} \to \mathbb{R}^2, \quad \sigma_N(x, y, z) = \Big(\frac{x}{1-z}, \frac{y}{1-z}\Big),$$

$$\sigma_S : \mathbb{S}^2 \setminus \{S\} \to \mathbb{R}^2, \quad \sigma_S(x, y, z) = \Big(\frac{x}{1+z}, \frac{y}{1+z}\Big).$$

The transition map is smooth (in fact, inversion on $\mathbb{R}^2 \setminus \{0\}$).

Tangent Space

Definition

We denote $C^\infty(M)$ as the collection of smooth functions $f : M \to \mathbb{R}$.

Definition (Tangent vectors via derivation)

A linear map $v : C^\infty(M) \to \mathbb{R}$ is called a **derivation** at $p \in M$ if it satisfies Leibniz rule:

$$v(fg) = f(p)v(g) + g(p)v(f), \quad \forall f, g \in C^\infty(M).$$

The collection of all derivations at a point p forms a vector space T_pM which we call the tangent space to M at p.

Definition (Tangent vectors via curves)

Two smooth curves $\alpha_1, \alpha_2 : (-\epsilon, \epsilon) \to M$ with $\alpha_1(0) = \alpha_2(0) = p$ are equivalent if their derivatives in any chart agree at 0. A **tangent vector** $v \in T_pM$ is an equivalence class $[\alpha]$.

Curves Act on Functions (Derivations)

Equivalence of definitions

Let $v = [\alpha] \in T_p\mathcal{M}$ be represented by a smooth curve $\alpha : (-\epsilon, \epsilon) \to \mathcal{M}$ with $\alpha(0) = p$. Then v defines the directional derivative (a *derivation* at p) by

$$v(f) \;:=\; (f \circ \alpha)'(0), \quad \forall f \in C^\infty(M).$$

This is well-defined (independent of the representative curve) and satisfies linearity and the Leibniz rule.

Local Coordinate Representation

Let (U, φ) be a chart with $p \in U$ and $a := \varphi \circ \alpha$. Then

$$(f \circ \alpha)'(0) = (f \circ \varphi^{-1} \circ \varphi \circ \alpha)'(0) = \sum_{i=1}^{n} a_i'(0) \frac{\partial}{\partial x^i}\bigg|_{\varphi(p)} f \circ \varphi^{-1}.$$

Under the basis given by φ, a tangent vector is the usual velocity vector $a'(0) \in \mathbb{R}^n$.

Curves Act on Functions (Derivations)

Local Coordinates View

This shows that in local coordinates (i.e. we work with $f \circ \varphi^{-1}$), we have the equivalence

$$v = \sum_{i=1}^{n} a_i'(0) \frac{\partial}{\partial x^i}\Big|_{\varphi(p)}$$

Euclidean Space

Here we see how the previous abstract definitions relate to the familar Euclidean setting.

$T_x\mathbb{R}^n$

Consider, the derivative operator $\left.\frac{\partial}{\partial x^i}\right|_x$ for $x \in \mathbb{R}^n$.

- We can consider the derivative $\left.\frac{\partial}{\partial x^i}\right|_x$ as a derivation over $C^\infty(\mathbb{R}^n)$ by the usual product rule.
- We can view this as a smooth curve α as well: let $\alpha(t) = x + te_i$ where $\{e_i\}$ are the standard basis vectors in $\mathbb{R}^n$. Then we have

$$[\alpha](f) = (f \circ \alpha)'(0) = \left.\frac{d}{dt}\right|_{t=0} f(x + te_i) = \left.\frac{\partial}{\partial x^i}\right|_x (f)$$

Euclidean Space

Basis for $T_x\mathbb{R}^n$

Further, it is easy to see that the standard Euclidean derivative operators $\left\{ \frac{\partial}{\partial x^i}\Big|_x \right\}_{i=1}^n$ form a basis for the tangent space $T_x\mathbb{R}^n$ in Euclidean space:

$$(f \circ \alpha)'(0) = \sum_{i=1}^{n} \alpha_i'(0) \frac{\partial}{\partial x^i}\Big|_x f$$

- We see that $\alpha'(0)$ represents $[\alpha]$ in this standard basis.

Intuition

The abstract definitions of tangent vector are to formalise the notion of directional derivatives in a coordinate-free way on manifolds.

Differential

Differential

For a smooth map $F : M \to N$, the differential $dF_p : T_pM \to T_{F(p)}N$ at a point $p \in M$ is defined by

$$dF_p([\alpha]) := [F \circ \alpha] \in T_{F(p)}N.$$

Derivation

For $g \in C^\infty(N)$, we have

$$dF_p([\alpha])(g) = [\alpha](g \circ F).$$

Some basic properties:

- dF_p is a linear map
- We have the chain rule: $d(G \circ F)_p = dG_{F(p)} \circ dF(p)$
- If F is a diffeomorphism, then dF_p is an isomorphism and $(dF_p)^{-1} = dF^{-1}_{F(p)}$.

Local Coordinates: Canonical Basis of T_pM

Fix a chart (U, φ) around p with coordinates $x = (x^1, \ldots, x^n) := \varphi(p)$.

- We note that $d\varphi_p$ is an isomorphism as φ is a diffeomorphism.

Coordinate vectors from the chart map

We define the canonical basis of T_pM given by φ as

$$\left.\frac{\partial}{\partial x^i}\right|_p := d(\varphi^{-1})_{\varphi(p)} \left(\left.\frac{\partial}{\partial x^i}\right|_{\varphi(p)} \right) \in T_pM.$$

Equivalently, as a derivation acting on $f \in C^\infty(\mathcal{M})$,

$$\left.\frac{\partial}{\partial x^i}\right|_p (f) = \left.\frac{\partial}{\partial x^i}\right|_{\varphi(p)} \left(f \circ \varphi^{-1} \right).$$

Local Coordinates: Canonical Basis of T_pM

Expansion of a tangent vector

Any $v \in T_pM$ can be written uniquely as

$$v = \sum_{i=1}^{n} v^i \left.\frac{\partial}{\partial x^i}\right|_p, \qquad v^i := v(x^i) = v(\varphi^i)$$

Notation

We also use the shorthand:

$$\partial_i|_p = \left.\frac{\partial}{\partial x^i}\right|_p$$

Why this matters in ML

This is the concrete interface between "abstract" objects (T_pM) and tensors/arrays ($\mathbb{R}^n$).

Local Coordinates

Conversion from curves to local coordinates

We see under the canonical basis of T_pM given by the chart (U, φ), we have the natural association of a smooth curve $\alpha : (-\epsilon, \epsilon) \to M$ given by

$$\alpha \mapsto (\varphi \circ \alpha)'(0) \in \mathbb{R}^n$$

under the canonical basis $\left\{ \left. \frac{\partial}{\partial x^i} \right|_p \right\}$.

- There also exists a natural inverse which establishes an isomorphism between the curve representation and T_pM.

Einstein Notation

Note

To simplify notation, **Einstein notation** is often used. For an expression involving indices (e.g. for basis elements etc.), the use of repeated indices for different terms corresponds to summing over those indices.

- For example:

$$v^{i,j}\partial_i|_p \text{ corresponds to } \sum_{i=1}^{n} v^{i,j}\partial_i|_p.$$

Working in Coordinates: Differentials

Euclidean Differential

Let $F : \mathbb{R}^n \to \mathbb{R}^m$ be a smooth function. We use x to denote coordinates on the domain and y to denote coordinates on the codomain. Under the standard bases of $T_p\mathbb{R}^n$ and $T_{F(p)}\mathbb{R}^m$, we compute the matrix representation of dF_p by

$$\begin{aligned} dF_p\left(\left.\frac{\partial}{\partial x^i}\right|_p\right)(f) &= \left.\frac{\partial}{\partial x^i}\right|_p (f \circ F) \\ &= \left.\frac{\partial f}{\partial y^j}\right|_{F(p)} \left.\frac{\partial F^j}{\partial x^i}\right|_p \\ &= \left(\left.\frac{\partial F^j}{\partial x^i}\right|_p \left.\frac{\partial f}{\partial y^j}\right|_{F(p)}\right)(f) \end{aligned}$$

where $f \in C^\infty(\mathbb{R}^m)$ and the second line uses the standard chain rule.

Working in Coordinates: Differentials

Euclidean Differential

Therefore, the matrix representation under the standard Euclidean bases is given by

$$\left(\left.\frac{\partial F^i}{\partial x^j}\right|_p\right) = \begin{pmatrix} \left.\frac{\partial F^1}{\partial x^1}\right|_p & \cdots & \left.\frac{\partial F^1}{\partial x^n}\right|_p \\ \vdots & & \vdots \\ \left.\frac{\partial F^m}{\partial x^1}\right|_p & \cdots & \left.\frac{\partial F^m}{\partial x^n}\right|_p \end{pmatrix}$$

We see that this is the familiar form of the Jacobian.

Working in Coordinates: Differentials

Manifold Differential

Let $F : M \to N$ be a smooth function and $p \in M$. Taking coordinate charts (U, φ) about p and (V, ψ) about $F(p)$, let $\hat{F} = \psi \circ F \circ \varphi$. Under the standard bases of T_pM and $T_{F(p)}N$ under these charts, we compute the matrix representation of dF_p by

$$\begin{aligned} dF_p\left(\left.\frac{\partial}{\partial x^i}\right|_p\right) &= dF_p\left(d(\varphi^{-1})_{\varphi(p)}\left.\frac{\partial}{\partial x^i}\right|_{\varphi(p)}\right) \\ &= d(\psi^{-1})_{\hat{F}(\varphi(p))}\left(d\hat{F}_{\varphi(p)}\left(\left.\frac{\partial}{\partial x^i}\right|_{\varphi(p)}\right)\right) \end{aligned}$$

where we use $F \circ \varphi^{-1} = \psi^{-1} \circ \hat{F}$.

Working in Coordinates: Differentials

Manifold Differential

$$dF_p\left(\left.\frac{\partial}{\partial x^i}\right|_p\right) = d(\psi^{-1})_{\hat{F}(\varphi(p))}\left(\left.\frac{\partial \hat{F}^j}{\partial x^i}\right|_{\varphi(p)}\left.\frac{\partial}{\partial y^j}\right|_{\hat{F}(\varphi(p))}\right)$$
$$= \left.\frac{\partial \hat{F}^j}{\partial x^i}\right|_{\varphi(p)}\left.\frac{\partial}{\partial y^j}\right|_{F(p)}$$

where we use the form of the Euclidean differential and the definition of the canonical basis.

- We see that the matrix representation of dF_p has the same structure. Indeed, the definition of dF_p is to provide a coordinate-independent meaning to the Jacobian.

Working in Coordinates: Change of Coordinates

Change of Coordinates

Let $(U, \varphi), (V, \psi)$ be charts on M such that $U \cap V \neq \emptyset$ and $p \in U \cap V$.

- We use the notation for $\psi \circ \varphi^{-1} : \varphi(U \cap V) \to \psi(U \cap V)$:

$$\psi \circ \varphi^{-1}(x) = (\tilde{x}^1(x), \ldots, \tilde{x}^n(x))$$

To express the cannonical basis given by φ in terms of the basis given by ψ, we note that

$$\begin{aligned} \left.\frac{\partial}{\partial x^i}\right|_p &= d(\varphi^{-1})_{\varphi(p)} \left(\left.\frac{\partial}{\partial x^i}\right|_{\varphi(p)} \right) \\ &= d(\psi^{-1})_{\psi(p)} \circ d(\psi \circ \varphi^{-1})_{\varphi(p)} \left(\left.\frac{\partial}{\partial x^i}\right|_{\varphi(p)} \right) \end{aligned}$$

Working in Coordinates: Change of Coordinates

Change of Coordinates

$$\begin{aligned}\left.\frac{\partial}{\partial x^i}\right|_p &= d(\psi^{-1})_{\psi(p)}\left(\left.\frac{\partial \tilde{x}^j}{\partial x^i}\right|_{\varphi(p)}\left.\frac{\partial}{\partial \tilde{x}^j}\right|_{\psi(p)}\right) \\ &= \left.\frac{\partial \tilde{x}^j}{\partial x^i}\right|_{\varphi(p)}\left.\frac{\partial}{\partial \tilde{x}^j}\right|_p,\end{aligned}$$

where we use the action of $d(\psi \circ \varphi^{-1})_{\varphi(p)}$ on basis elements from above.

Working in Coordinates: Change of Coordinates

Note

Therefore, for some $v \in T_pM$, we have the representations:

$$v = v^i \frac{\partial}{\partial x^i}\bigg|_p = \tilde{v}^i \frac{\partial}{\partial \tilde{x}^i}\bigg|_p,$$

the expression for the change of coordinates gives:

$$\tilde{v}^j = \frac{\partial \tilde{x}^j}{\partial x^i}\bigg|_{\varphi(p)} v^i$$

from evaluating the LHS with ψ^j (recall the action of the standard basis vectors on the component functions of the underlying chart).

Submanifolds

Smooth Embedding

Let M, N be smooth manifolds. A **smooth embedding** of M into N is a smooth mapping $F : M \to N$ such that dF_p is injective for all $p \in M$ and F is a homeomorphism onto its image $F(M) \subset N$ under the subspace topology.

Embedded Submanifolds

In the case of a smooth manifold $S \subset \mathbb{R}^n$, the inclusion map $\iota : S \hookrightarrow \mathbb{R}^n$ provides a smooth embedding.

- Moreover, we have the identification of T_pS as a subspace of $\mathbb{R}^n \cong T_p\mathbb{R}^n$ as $d\iota_p$ is injective.

Submanifolds

Sphere

Consider $\mathbb{S}^2$ and a smooth curve $\alpha : (-\epsilon, \epsilon) \to \mathbb{S}^2$ such that $\alpha(0) = p$. We can view this curve in $\mathbb{R}^3$ by the inclusion map: $\gamma = \iota \circ \alpha$.

- We have $\|\gamma\|_2^2 = 1$ so by differentiating both sides with $t = 0$, we get the constraint:

$$p^\top \dot{\gamma}(0) = 0,$$

- Under the standard basis of $T_p\mathbb{R}^n$, we see $\dot{\gamma}(0) \in T_p\mathbb{R}^n$.
- This allows us to characterise the tangent space of $\mathbb{S}^2$:

$$T_p\mathbb{S}^2 = \{v \in \mathbb{R}^3 : p^\top v = 0\},$$

which aligns with our intuition.

The Tangent Bundle TM as a Smooth Manifold

Definition

The **tangent bundle** is the disjoint union

$$TM := \bigsqcup_{p \in M} T_p M, \qquad \pi : TM \to M,\ \pi(p, v) = p.$$

The tangent bundle has a natural smooth manifold structure with $\dim TM = 2n$.

Charts on TM (construction)

Given a chart (U, φ) on M, define a map

$$\widetilde{\varphi} : \pi^{-1}(U) \to \varphi(U) \times \mathbb{R}^n, \qquad \widetilde{\varphi}\left(\sum_{i=1}^{n} v^i \frac{\partial}{\partial x^i}\bigg|_p \right) = \Big(\varphi(p),\ v \Big).$$

The inverse has the form $(x, v) \mapsto \sum_{i=1}^{n} v^i \frac{\partial}{\partial x^i}\Big|_{\varphi^{-1}(x)}$. Moreover, the transition map involves the change-of-coordinates etc. which is smooth.

Vector Fields

Vector Fields

A **vector field** is a smooth section $X : M \to TM$—i.e. $\pi \circ X = \mathrm{id}$ and X is a smooth function.

- Given a chart (U, φ), we can represent X at $p \in U$ in terms of the canonical basis:

$$X_p = X^i(p) \frac{\partial}{\partial x^i}\bigg|_p,$$

where we call $X^i : U \to \mathbb{R}$ the **component functions** of X.

Note

X is smooth iff for all charts, the resulting component functions X^i are smooth.

Vector Fields

Note

Let $f \in C^\infty(M)$, we define $Xf : M \to \mathbb{R}$ as the function where for each $p \in M$, we apply $X_p \in T_pM$ to f.

- Xf is a smooth function in $C^\infty(M)$
- Moreover, this allows X to define the map (derivation): $X : C^\infty(M) \to C^\infty(M)$. This mapping is linear and satisfies:

$$X(fg) = fXg + gXf, \qquad \forall f, g \in C^\infty(M).$$

- There is a bijective relationship between derivations and smooth vector fields.

Local Frames

Local Frames

Let $(E_1, \ldots, E_n)$ be a ordered tuple of vector fields defined on some open subset $U \subset M$.

- We say $(E_1, \ldots, E_1)$ is a **local frame** for M if for all $p \in U$, the vectors:

$$(E_1|_p, \ldots, E_n|_p)$$

  forms a basis for T_pM.

Flows

- An **integral curve** through p is smooth curve $\gamma(t)$ satisfying

$$\dot{\gamma}(t) = X(\gamma(t)), \qquad \gamma(0) = p.$$

- (Local) existence/uniqueness holds in charts by reducing to an ODE on $\mathbb{R}^n$.
- The associated **flow** is a map $\Phi_t : M \to M$ such that $\Phi_0 = \mathrm{id}$ and

$$\frac{\mathrm{d}}{\mathrm{d}t}\Phi_t(p) = X(\Phi_t(p)).$$

Time-dependent vector fields

For X_t, the flow solves $\dot{\gamma}(t) = X_t(\gamma(t))$. This is the basic object behind **flow matching** and **continuous normalizing flows** on manifolds.

Riemannian Metric

A **Riemannian Metric** g on a smooth manifold M is not a distance function $d(x, y)$. Instead, it is a smooth assignment of an **inner product** to the tangent space T_pM at every point p.

Definition

For each $p \in M$, g_p is a map:

$$g_p : T_pM \times T_pM \to \mathbb{R}$$

satisfying three properties for all $u, v \in T_pM$:

1. **Bilinear:** Linear in both arguments.
2. **Symmetric:** $g_p(u, v) = g_p(v, u)$.
3. **Positive Definite:** $g_p(u, u) \geq 0$, with equality iff $u = 0$.

Riemannian Metric

Note

The smooth assignment is due to a smooth section: $M \to S^2T^*M$ where S^2T^*M denotes the bundle of symmetric covariant 2-tensors on M.

- S^2T^*M can be given a smooth structure (in a similar way as TM)

Local Coordinates

Local Coordinates

In local coordinates given by a chart (U, φ), we have the basis $\{\partial_i|_p\}_{i=1}^n$.

- We have

$$g_p(u,v) = g_p\left(\sum_{i=1}^n u^i \partial_i|_p, \sum_{j=1}^n v^j \partial_j|_p\right) = \sum_{i,j} u^i v^j g_{ij} = u^\top G v$$

where $G = (g_{ij})$ and $g_{ij} = g_p(\partial_i|_p, \partial_j|_p)$. Additionally, G is a symmetric positive definite matrix.

Induced Metric

Induced Metric

For a Riemannian manifold (M, g), let S be a (embedded) submanifold with inclusion $\iota : S \hookrightarrow M$.

$$(\iota^* g)_p(u, v) = g_p(d\iota_p u, d\iota_p v).$$

From our identification of $T_p S$ as a subspace of $T_p M$, this can be considered the restriction of g to vectors tangent to S.

Riemannian Geometry

Geometric Quantities

Given a Riemannian metric g, we can now define the following geometric quantities on smooth manifolds:

- **Vector Norms**: $\|v\|_{g_p} = \sqrt{g_p(v,v)}$ for all $v \in T_pM$
- **Length**: For a smooth curve $\gamma : [a,b] \to M$, we define the length as

$$L(\gamma) = \int_a^b \|\gamma'(t)\|_{g_{\gamma(t)}} \, dt$$

This allows to define the notion of *straight* paths on smooth manifolds: we term paths that minimise the distance between two points on M a **geodesic**.

Example: $\mathbb{S}^2$

On $\mathbb{S}^2$ (under the induced metric), the geodesics corresponds to the great circles.

Local Orthonormal Frames

Note

For a Riemannian manifold (M, g), it can be shown that for each $p \in M$, there exists a smooth orthonormal frame on a neighbourhood of p.

- Start with the local frame defined by a chart about p and apply Gram-Schmidt to the canonical coordinate basis.

Riemannian Gradient

Abstract definition

For $f \in C^\infty(\mathcal{M})$, the **Riemannian gradient** $\nabla_p f \in T_p\mathcal{M}$ is defined by

$$g_p(\nabla_p f, v) = \mathrm{d}f_p(v) \quad \forall v \in T_p\mathcal{M}.$$

Steepest Descent

If we want to maximise $df_p(v)$ (i.e. to find the direction of steepest descent) with the constraint $\|v\|_g = 1$, by definition this is equivalent to maximising the LHS:

$$|g_p(\nabla_p f, v)| \leq \|\nabla_p f\|_g \|v\|_g ,$$

as this is an inner product, we can apply Cauchy-Schwarz to conclude we need $v \propto \nabla_p f$.

Riemannian Gradient

Coordinate formula

Considering local coordinates given by the chart (U, φ), we have $v = \sum_{i=1}^{n} v^i \partial_i|_p$ and $\nabla_p f = \sum_{i=1}^{n} u^i \partial_i|_p$. This gives

$$g_p(\nabla_p f, v) = \sum_{i,j=1}^{n} v^i u^j g_{ij},$$

and

$$df_p(v) = d(f \circ \varphi^{-1}) \circ d\varphi_p \left(\sum_{i=1}^{n} v^i \partial_i|_p \right) = \sum_{i=1}^{n} v^i \partial_i|_{\varphi(p)} (f \circ \varphi^{-1}),$$

by the definition of $\partial_i|_p$ and the form of the Euclidean differential as $f \circ \varphi^{-1} : \mathbb{R}^n \to \mathbb{R}$.

Riemannian Gradient

Coordinate formula

Therefore, we have

$$\begin{aligned}
\sum_{i,j=1}^{n} v^i u^j g_{ij} &= v^i \partial_i|_{\varphi(p)}(f \circ \varphi^{-1}) \\
\implies \sum_{j=1}^{n} u^j g_{ij} &= \partial_i|_{\varphi(p)}(f \circ \varphi^{-1}) \\
Gu &= \nabla_{\varphi(p)}(f \circ \varphi^{-1}) \\
\implies u &= G^{-1}\nabla_{\varphi(p)}(f \circ \varphi^{-1}).
\end{aligned}$$

where we use $G = (g_{ij})$ and $G^\top = G$. This explains the usual form of the Riemannian gradient in applications (e.g. natural gradient descent).

Riemannian Gradient

Note

The above proof shows that in local coordinates $\nabla_p f$ has smooth coefficients (by definition g_{ij} are smooth), hence, $p \mapsto \nabla_p f$ defines a smooth vector field.

Riemannian Gradient: Chain Rule

Scalar chain rule

Let $f \in C^\infty(\mathcal{M})$ and let $h : \mathbb{R} \to \mathbb{R}$ be smooth. Since the differential obeys the usual chain rule,

$$\mathrm{d}(h \circ f) = h'(f)\,\mathrm{d}f.$$

By definition of the Riemannian gradient, the following holds for all $v \in T_pM$

$$g(\nabla(h \circ f), v) = \mathrm{d}(h \circ f)(v) = h'(f)\,\mathrm{d}f(v) = h'(f)\,g(\nabla f, v) = g(h'(f)\nabla f, v)$$

Nondegeneracy of g implies

$$\boxed{\nabla(h \circ f) = h'(f)\,\nabla f} \qquad \text{(pointwise on } \mathcal{M}\text{)}.$$

Riemannian Gradient: Chain Rule

Example: $h(t) = \log t$

If $f > 0$ is smooth, then $h'(t) = 1/t$ and therefore

$$\boxed{\nabla(\log f) = \tfrac{1}{f}\,\nabla f}.$$

In local coordinates this is completely explicit:

$$(\nabla \log f)_{\text{coords}} = G(p)^{-1}\,\nabla_p(\log f) = G(p)^{-1}\,\frac{1}{f(p)}\nabla_p f = \frac{1}{f(p)}(\nabla f)_{\text{coords}}.$$

This will be useful later for the Riemannian score matching objective.

Exponential Map

Let (M, g) be a Riemannian manifold and $p \in M$.

Geodesic Existence

For every tangent vector $v \in T_pM$, there exists a unique geodesic $\gamma_v(t)$ satisfying:

1. $\gamma_v(0) = p$ (Starts at p)
2. $\gamma_v'(0) = v$ (Initial velocity is v)

Exponential Map

The **exponential map** $\exp_p : T_pM \to M$ is defined as the point reached at time $t = 1$:

$$\exp_p(v) := \gamma_v(1)$$

Note: The map is generally only defined on a neighbourhood of $0 \in T_p\mathcal{M}$, not necessarily the whole space.

Exponential Map

Sphere

Consider $\mathbb{S}^2$ and let $p \in \mathbb{S}^2, v \in T_p\mathbb{S}^2$ with the standard identification in $\mathbb{R}^3$.

- The geodesics are **great circles**. The unique curve starting at p with velocity v is given by rotating p in the plane spanned by $\{p, v\}$.
- We have the closed form:

$$\exp_p(v) = \cos(\|v\|)\, p + \sin(\|v\|)\, \frac{v}{\|v\|}$$

Exponential Map

Matrix Lie Groups

Let $G \subset GL(n, \mathbb{R})$ be a matrix Lie Group (e.g., rotation matrices $SO(n)$).

- The tangent space at identity T_eG is the **Lie algebra** $\mathfrak{g}$.
- Elements $X \in \mathfrak{g}$ are matrices.

In this case, the Riemannian exponential (for a bi-invariant metric) coincides with the standard **matrix exponential**:

$$\exp_e(X) = e^X = \sum_{k=0}^{\infty} \frac{X^k}{k!} = I + X + \frac{X^2}{2} + \dots$$

Geodesics: The geodesics through identity are **one-parameter subgroups**:

$$\gamma(t) = e^{tX}$$

Probability on Manifolds

- On an oriented manifold, we integrate **top-degree forms**.
- With a Riemannian metric g, there is a canonical volume form (the **Riemannian volume element**).

In local coordinates

If g has matrix $[g_{ij}(x)]$ in a chart, then

$$\mathrm{dVol}_g = \sqrt{|\det g(x)|}\, \mathrm{d}x^1 \cdots \mathrm{d}x^n.$$

Probability densities

A density p on $\mathcal{M}$ is defined relative to dVol_g:

$$\mathbb{P}(A) = \int_A p(x)\, \mathrm{dVol}_g(x)$$

Riemannian Score-Based Generative Modelling

Valentin De Bortoli[*,†], **Émile Mathieu**[*,‡], **Michael Hutchinson**[*,‡]

James Thornton[‡], **Yee Whye Teh**[‡], **Arnaud Doucet**[‡]

Abstract

Score-based generative models (SGMs) are a powerful class of generative models that exhibit remarkable empirical performance. Score-based generative modelling (SGM) consists of a "noising" stage, whereby a diffusion is used to gradually add Gaussian noise to data, and a generative model, which entails a "denoising" process defined by approximating the time-reversal of the diffusion. Existing SGMs assume that data is supported on a Euclidean space, i.e. a manifold with flat geometry. In many domains such as robotics, geoscience or protein modelling, data is often naturally described by distributions living on Riemannian manifolds and current SGM techniques are not appropriate. We introduce here *Riemannian Score-based Generative Models* (RSGMs), a class of generative models extending SGMs to Riemannian manifolds. We demonstrate our approach on a variety of manifolds, and in particular with earth and climate science spherical data.

Riemannian Score-Based Generative Modelling

Generalising diffusion models to general Riemannian manifolds requires thinking about:

- How do we define SDEs on manifolds?
- Do we have an equivalent of the score matching objective on manifolds?
- How do we parametrise the score function (remember that this should live in the tangent space)?
- How do we sample from SDEs on manifolds?

SDEs on Manifolds

The general theory of SDEs on manifolds is complex (Hsu, 2002)

- We will not go in depth into the theory.

Instead, we will consider the setting of **compact** manifolds (think closed and bounded sets). This includes most cases of interest: $\mathrm{SO}(3), \mathbb{S}^2$ etc.

Brownian Motion

In order to define a suitable noising process, **Brownian motion** is a natural choice.

- As we consider compact manifolds M, we have access to a uniform distribution on M given by $\mu = \frac{\mathrm{dVol}_g}{|M|}$ where $|M| = \int_M \mathrm{dVol}_g(x)$.
- If we noise p_{data} with Brownian motion, we will converge to the uniform distribution.

Markov Semigroups

From the theory of Markov semigroups, we can define stochastic processes on general spaces via their **infinitesimal generator** $L : C^\infty(M) \to C^\infty(M)$.

- In particular, the transition density $p_{t|0}(x_t|x_0)$ (w.r.t. the uniform measure on M) is given by the Kolmogorov forward equation:

$$\frac{d}{dt}p_{t|0}(x_t|x_0) = L^* p_{t|0}(x_t|x_0), \qquad p_{0|0}(x_t|x_0) = \delta_{x_0}(x_t),$$

  and L^* denotes the adjoint operator to L—i.e. $\langle Lf, g\rangle = \langle f, L^* g\rangle$.

- Additionally, given the initial condition $p_0 = p_{\text{data}}$, the marginal distributions also follow:

$$\frac{d}{dt}p_t(x_t) = L^* p_t(x_t), \qquad p_0 = p_{\text{data}}.$$

See the notes on my website for an introduction to semigroups and generators (and discrete diffusion).

Laplace-Beltrami Operator

Generator of Brownian Motion

Brownian motion in $\mathbb{R}^n$ is defined by the choice:

$$L = \frac{1}{2}\Delta,$$

where Δ denotes the Laplacian. Note that the Laplacian is self-adjoint.

Laplace-Beltrami

The generalisation of the Laplacian to Riemannian geometry is called the **Laplace-Beltrami** operator Δ_M:

$$\Delta_M f = \mathrm{div}(\nabla f),$$

where div denotes the Riemannian divergence. This is also self-adjoint.

- This operator defines the generalisation of Brownian motion to smooth manifolds.

Time-Reversal on Riemannian Manifolds

Time-Reversal Theorem (simplified)

Let $(X_t)_{t\in[0,T]}$ be the stochastic process defined by the (forward) SDE:

$$dX_t = dB_t^M,$$

where this denotes the SDE defined by the generator $\frac{1}{2}\Delta_M$. We assume that the law of X_t admits a smooth positive density p_t (with respect to μ).

- Let $(Y_t)_{t\in[0,T]} = (X_{T-t})_{t\in[0,T]}$, then $(Y_t)_{t\in[0,T]}$ is the stochastic process defined by the (reverse) SDE:

$$dY_t = \nabla \log p_{T-t}(X_t)dt + dB_t^M.$$

Transition Densities

It is known in the literature that the transition densities $p_{t|0}$ have the form:

$$p_{t|0}(x_t|x_0) = \sum_{i=1}^{\infty} e^{-\lambda_i t}\phi_i(x_t)\phi_i(x_0),$$

where λ_i, ϕ_i are the eigenvalues and eigenvectors of $-\Delta_M$ respectively.

Note

We can approximate $\nabla_{x_t} \log p_{t|0}(x_t|x_0)$ by using the truncated sum:

$$\nabla_{x_t} \log p_{t|0}(x_t|x_0) \approx \nabla_{x_t} \log \sum_{i=1}^{I} e^{-\lambda_i t}\phi_i(x_t)\phi_i(x_0)$$

Transition Densities

$\mathbb{S}^2$

In the case of the sphere:

- **Eigenvalues**: these are given by $\{k(k+1) : k \in \mathbb{N}\}$.
- **Eigenvectors**: these are given by spherical harmonics.

The form of $p_{t|0}$ can be further simplified via Gegenbauer polynomials.

Riemannian Denoising Score Matching

Under standard regularity assumptions, we can interchange $\int$ and ∇.

- It is easy to recover the score matching identity:

$$\begin{aligned}\nabla_{x_t} \log p_t(x_t) &= \int_M \nabla_{x_t} \log p_{t|0}(x_t|x_0) p_{0|t}(x_0|x_t) d\mu(x_0) \\ &= \mathbb{E}_{x_0|x_t}[\nabla_{x_t} \log p_{t|0}(x_t|x_0)],\end{aligned}$$

  due to the Riemannian gradient chain rule and the fact we have densities (wrt μ).

- To recover an analogous score matching objective, we need to show that $\nabla_{x_t} \log p_t(x_t)$ minimises:

$$\ell(s) = \int_M \left\| s - \nabla_{x_t} \log p_{t|0}(x_t|x_0) \right\|_g^2 p_{0|t}(x_0|x_t) d\mu(x_0).$$

Riemannian Denoising Score Matching

This can be achieved with the standard approach to showing expectations minimise L2 losses.

$$\implies \text{expand } \mathbb{E}_{x_0|x_t}\left[\left\|s - \nabla_{x_t}\log p_t(x_t) + \nabla_{x_t}\log p_t(x_t) - \nabla_{x_t}\log p_{t|0}(x_t|x_0)\right\|_g^2\right]$$

The main term to analyse is

$$\mathbb{E}_{x_0|x_t}\left[g_{x_t}\left(s - \nabla_{x_t}\log p_t(x_t), \nabla_{x_t}\log p_t(x_t) - \nabla_{x_t}\log p_{t|0}(x_t|x_0)\right)\right].$$

We want to show this term is zero to conclude the score matching objective.

Note

To justify pulling $s - \nabla_{x_t}\log p_t(x_t)$ outside of the integral, we note that we can always express the terms via an orthonormal basis of $T_{x_t}M$ (left as an exercise...)

Score Parametrisation

The score $\nabla_{x_t} \log p_t(x_t)$ is a vector field $M \to TM$ (for each time t). We have the following parametrisations for our learnt score approximation s_θ:

- **Projected Vector Field**: Define $s_\theta(t, x) = \text{proj}_{T_x M}(\tilde{s}_\theta(t, x))$ where $\tilde{s}_\theta : [0, T] \times \mathbb{R}^N \to \mathbb{R}^N$ is an ambient vector field and $\text{proj}_{T_x M}$ denotes the projection to the subspace $T_x M$ (this is given by a projection matrix).
- This does assume that we are embedding M within an ambient Euclidean space.

See the paper for other choices of parametrisations.

Sampling

Any "well-behaved" diffusion process on M can be approximated by **Geodesic Random Walks** (Jørgensen, 1975).

(a) A single step of a Geodesic Random Walk.

(b) Many steps yield an approximate trajectory.

(c) Gaussian Random Walk [Left] and the Brownian motion density [Right] agree well for small time steps.

Algorithm 1 GRW (Geodesic Random Walk)

Require: $T, N, X_0^\gamma, b, \sigma, \mathrm{P}$

$\gamma = T/N$ ▷ Step-size

for $k \in \{0, \dots, N-1\}$ **do**

  $Z_{k+1} \sim \mathrm{N}(0, \mathrm{Id})$ ▷ Sample a Gaussian in the tangent space of X_k^γ

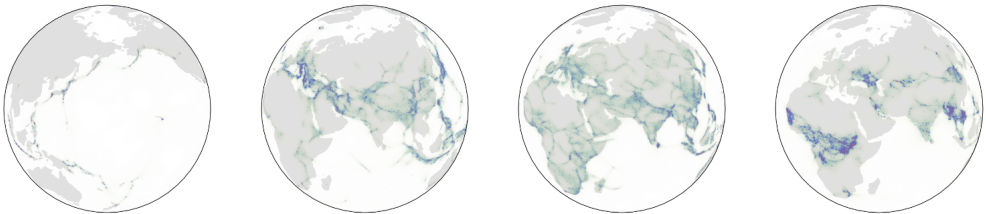
  $W_{k+1} = \gamma b(k\gamma, X_k^\gamma) + \sqrt{\gamma}\sigma(k\gamma, X_k^\gamma) Z_{k+1}$ ▷ Compute the Euler–Maruyama step on tangent space

  $X_{k+1}^\gamma = \exp_{X_k^\gamma}[W_{k+1}]$ ▷ Move along the geodesic defined by W_{k+1} and X_k^γ on $\mathcal{M}$

return $\{X_k^\gamma\}_{k=0}^N$

Experiments

Training on empirical distribution of occurrences of earth and climate science events on the surface of the earth (volcanic eruptions, earthquakes, floods, wild fires).



(a) Volcano (b) Earthquake (c) Flood (d) Fire

Figure 2: Trained score-based generative models on earth sciences data. The learned density is colored green-blue. Blue and red dots represent training and testing datapoints, respectively.

Modern Directions

SigmaDock

SigmaDock: Untwisting Molecular Docking With Fragment-Based SE(3) Diffusion

Alvaro Prat Leo Zhang Charlotte M. Deane Yee Whye Teh Garrett M. Morris
Department of Statistics, University of Oxford

SigmaDock

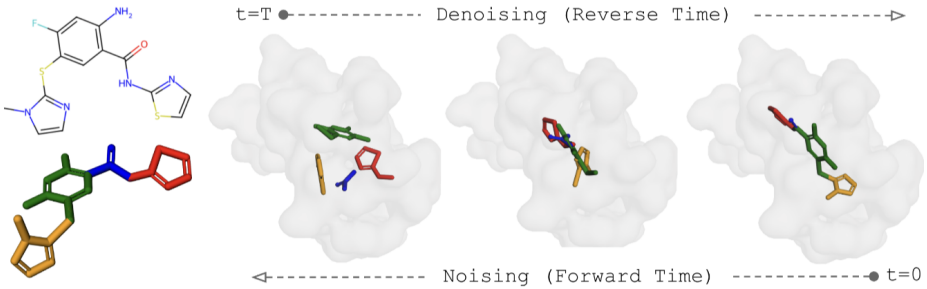


Figure 1: Illustration of SIGMADOCK using PDB 1V4S and ligand MRK. We create an initial conformation of a query ligand where we define our m rigid body fragments (colour coded). The corresponding forward diffusion process operates in $\mathrm{SE}(3)^m$ via independent roto-translations.

Architecture Design

Residual Connections

Deep Residual Learning for Image Recognition

Kaiming He Xiangyu Zhang Shaoqing Ren Jian Sun
Microsoft Research
{kahe, v-xiangz, v-shren, jiansun}@microsoft.com

Abstract

Deeper neural networks are more difficult to train. We present a residual learning framework to ease the training of networks that are substantially deeper than those used previously. We explicitly reformulate the layers as learning residual functions with reference to the layer inputs, instead of learning unreferenced functions. We provide comprehensive empirical evidence showing that these residual networks are easier to optimize, and can gain accuracy from considerably increased depth. On the ImageNet dataset we evaluate residual nets with a depth of up to 152 layers—8× deeper than VGG nets [41] but still having lower complexity. An ensemble of these residual nets achieves 3.57% error on the ImageNet test *set. This result won the 1st place on the ILSVRC 2015 classification task. We also present analysis on CIFAR-10 with 100 and 1000 layers.*

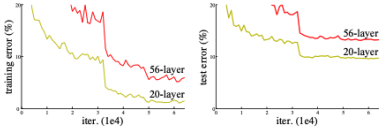


Figure 1. Training error (left) and test error (right) on CIFAR-10 with 20-layer and 56-layer "plain" networks. The deeper network has higher training error, and thus test error. Similar phenomena on ImageNet is presented in Fig. 4.

greatly benefited from very deep models.

Driven by the significance of depth, a question arises: *Is learning better networks as easy as stacking more layers?* An obstacle to answering this question was the notorious problem of vanishing/exploding gradients [1, 9], which

Representation Learning

From Ji et al. (2025)

Figure 4: Visualize learned representations from pretrained DINO models **without cherry-picking**.

Rank Collapse in Transformers

Attention is not *all* you need:
pure attention loses rank doubly exponentially with depth

Yihe Dong
Google
yihed@google.com

Jean-Baptiste Cordonnier
EPFL
jean-baptiste.cordonnier@epfl.ch

Andreas Loukas
EPFL
andreas.loukas@epfl.ch

Orthogonal Self-Attention

Orthogonal Self-Attention

Leo Zhang[1*] and James Martens[†]
[1] *Department of Statistics, University of Oxford*

Abstract

Softmax Self-Attention (SSA) is a key component of Transformer architectures. However, when utilised within skipless architectures, which aim to improve representation learning, recent work has highlighted the inherent instability of SSA due to inducing rank collapse and poorly-conditioned Jacobians. In this work, we design a novel attention mechanism: Orthogonal Self-Attention (OSA), which aims to bypass these issues with SSA, in order to allow for (non-causal) Transformers without skip connections and normalisation layers to be more easily trained. In particular, OSA parametrises the attention matrix to be orthogonal via mapping a skew-symmetric matrix, formed from query-key values, through the matrix exponential. We show that this can be practically implemented, by exploiting the low-rank structure of our query-key values, resulting in the computational complexity and memory cost of OSA scaling linearly with sequence length. Furthermore, we derive an initialisation scheme for which we prove ensures that the Jacobian of OSA is well-conditioned.

Orthogonal Self-Attention

Orthogonal Self-Attention

We define **Orthogonal Self-Attention (OSA)** as

$$\text{OSA}(X) = A(X)XW^VW^O \text{ where } A(X) = \exp(S) \text{ and } S = \frac{\alpha}{\sqrt{d_v}}\left(QK^\top - KQ^\top\right),$$

and $Q = XW^Q, K = XW^K$.

- $X \in \mathbb{R}^{N\times d}, W^Q, W^K, W^V \in \mathbb{R}^{d\times d_v}, W^O \in \mathbb{R}^{d_v\times d}$ and $d_v = d/h$ where h is the number of attention heads.

Low-Rank Trick

Theorem

Let $B(X) \in \mathbb{R}^{N \times r}$ be an orthonormal matrix where the columns provide a basis for the subspace U ($\dim U = r$) spanned by the columns of Q, K. Then we have

$$\exp(S(X)) = I_N + B(X)[\exp[S](X) - I_r]B(X)^\top,$$

where

$$[S](X) = B(X)^\top S(X) B(X) \in \mathbb{R}^{r \times r}.$$

Cost analysis

This allows for computational complexity and memory to scale linearly with N.

Other Considerations

- **How to compute** $B(X)$: QR / Newton-Schultz
- **Error analysis**: for Newton-Schultz convergence
- **Initialisation**: for W^Q, W^K, W^V, W^O and α and the MLPs

Experiments

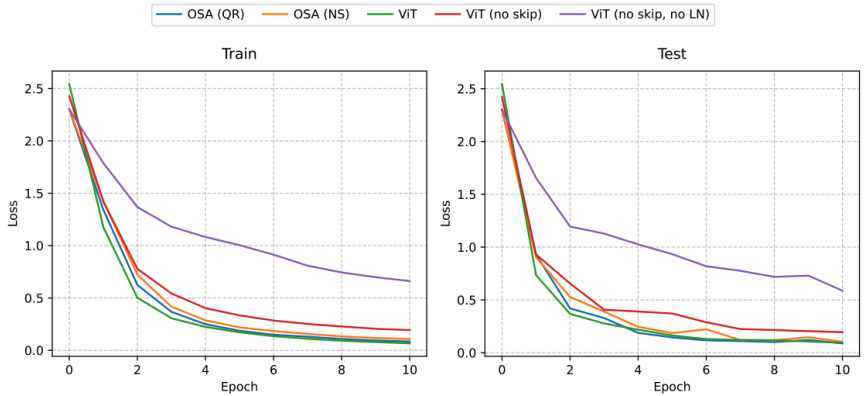


Figure 1: Train and test loss curves for OSA-Transformer and ViT models trained on MNIST for classification.

Optimisation

Steepest Descent

Old Optimizer, New Norm: An Anthology

Jeremy Bernstein JBERNSTEIN@MIT.EDU
Laker Newhouse LAKERN@MIT.EDU
MIT CSAIL, United States

Abstract

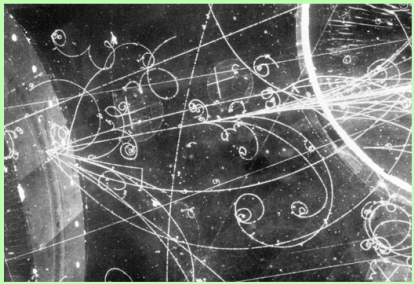
Deep learning optimizers are often motivated through a mix of convex and approximate second-order theory. We select three such methods—Adam, Shampoo and Prodigy—and argue that each method can instead be understood as a squarely first-order method without convexity assumptions. In fact, after switching off exponential moving averages, each method is equivalent to *steepest descent* under a particular *norm*. By generalizing this observation, we chart a new design space for training algorithms. Different operator norms should be assigned to different tensors based on the role that the tensor plays within the network. For example, while linear and embedding layers may have the same weight space of $\mathbb{R}^{m\times n}$, these layers play different roles and should be assigned different norms. We hope that this idea of carefully metrizing the neural architecture might lead to more stable, scalable and indeed faster training.

Muon

jeremybernste.in research teaching writing cv

Deriving Muon

Boston, 7 Mar 2025



Particle tracks in a bubble chamber. *Fermilab.*

We recently proposed *Muon*: a new neural net optimizer. Muon has garnered attention for its excellent practical performance: it was used to set NanoGPT speed records leading to interest from the big labs.

Thank you!

leo.zhang@stx.ox.ac.uk

https://leozhangml.github.io/

References I

Bloem-Reddy, B. and Teh, Y. W. (2020). Probabilistic symmetries and invariant neural networks. *Journal of Machine Learning Research*, 21(90):1–61.

Bronstein, M. M., Bruna, J., Cohen, T., and Veličković, P. (2021). Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*.

Chen, R. T. and Lipman, Y. (2023). Flow matching on general geometries. *arXiv preprint arXiv:2302.03660*.

Cornish, R. (2024). Stochastic neural network symmetrisation in markov categories. *arXiv preprint arXiv:2406.11814*.

Cornish, R., Caterini, A., Deligiannidis, G., and Doucet, A. (2020). Relaxing bijectivity constraints with continuously indexed normalising flows. In *International conference on machine learning*, pages 2133–2143. PMLR.

References II

De Bortoli, V., Mathieu, E., Hutchinson, M., Thornton, J., Teh, Y. W., and Doucet, A. (2022). Riemannian score-based generative modelling. *Advances in neural information processing systems*, 35:2406–2422.

Helfrich, K., Willmott, D., and Ye, Q. (2018). Orthogonal recurrent neural networks with scaled cayley transform. In *International Conference on Machine Learning*, pages 1969–1978. PMLR.

Hsu, E. P. (2002). *Stochastic analysis on manifolds*. Number 38. American Mathematical Soc.

Ji, Y., Martens, J., Zheng, J., Zhou, Z., Moghadam, P., Zhang, X., Saratchandran, H., and Lucey, S. (2025). Cutting the skip: Training residual-free transformers. *arXiv preprint arXiv:2510.00345*.

Jørgensen, E. (1975). The central limit problem for geodesic random walks. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 32(1):1–64.

References III

Lee, J. M. (2003). Smooth manifolds. In *Introduction to smooth manifolds*, pages 1–29. Springer.

Mathieu, E. and Nickel, M. (2020). Riemannian continuous normalizing flows. *Advances in neural information processing systems*, 33:2503–2515.

Wang, T. and Isola, P. (2020). Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pages 9929–9939. PMLR.

Xiao, L., Bahri, Y., Sohl-Dickstein, J., Schoenholz, S., and Pennington, J. (2018). Dynamical isometry and a mean field theory of cnns: How to train 10,000-layer vanilla convolutional neural networks. In *International conference on machine learning*, pages 5393–5402. PMLR.

References IV

Yim, J., Trippe, B. L., De Bortoli, V., Mathieu, E., Doucet, A., Barzilay, R., and Jaakkola, T. (2023). Se (3) diffusion model with application to protein backbone generation. *arXiv preprint arXiv:2302.02277*.

Zhang, L., Ashouritaklimi, K., Teh, Y. W., and Cornish, R. (2024). Symdiff: Equivariant diffusion via stochastic symmetrisation. *arXiv preprint arXiv:2410.06262*.

Zhang, L. and Martens, J. (2026). Orthogonal self-attention. *arXiv preprint arXiv:2602.05996*.