



データの幾何学と深層学習

～画像やテキストの構造を見つける～

松原 崇 (北海道大学/CyberAgent/理研AIP)

自己紹介

- ～2015: 大阪大学 基礎工学研究科 博士後期課程
 - 非線形力学系としての神経系と回路実装
- 2015～2019: 神戸大学 システム情報学研究科 助教
 - マルチメディア & 医療データ情報処理
 - 深層生成モデル
- 2020～2023: 大阪大学 基礎工学研究科 准教授
 - 解析力学 × 深層学習
 - コンピュータビジョン(幾何学的深層学習)
- 2024～ : 北海道大学 情報科学研究院 教授
 - 力学系 × 深層学習
 - コンピュータビジョン(データの幾何学)
- 2025～ : CyberAgent, AI Lab リサーチサイエンティスト
理化学研究所 革新知能統合研究センター(AIP) 客員研究員



はじめに

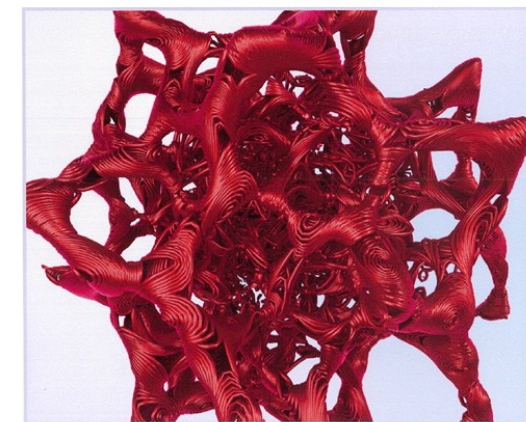
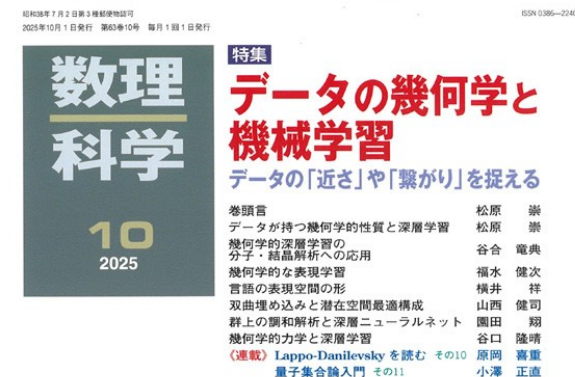
『数理科学』 2025年10月号 “データの幾何学と機械学習”

■ 特集概要

- データが持つ幾何学について深い洞察のある研究を集める

■ 記事

- 松原崇 “データが持つ幾何学的性質と深層学習”
- 谷合竜典 “幾何学的深層学習の分子・結晶解析への応用”
- 福水健次 “幾何学的な表現学習”
- 横井祥 “言語の表現空間の形”
- 山西健司 “双曲埋め込みと潜在空間最適構成”
- 園田翔 “群上の調和解析と深層ニューラルネット”
- 谷口隆晴 “幾何学的力学と深層学習”



サイエンス社

No.748

はじめに

“データの幾何学”という分野は(まだ)ない

- 散発的に論文や講演などがある程度
- しかも実際のフォーカスはバラバラ
 - グラフ
 - データ多様体
 - 幾何学的対称性
 - etc.

分野としてまとめてみよう

三

Proceedings of the AAAI Conference on Artificial Intelligence

Home / Archives /

Vol. 38 No. 20: AAAI-24 Special Track AI for Social Impact, Senior Member Presentations, New Faculty Highlights, Journal Track

/

New Faculty Highlights

Exploiting Data Geometry in Machine Learning

Melanie Weber

Harvard University

DOI: <https://doi.org/10.1609/aaai.v38i20.30297>

Keywords: Geometric Learning, Representation Learning, Graph Machine Learning, Riemannian Optimization, Algorithms

Abstract

A key challenge in Machine Learning (ML) is the identification of geometric structure in high-dimensional data. Most algorithms assume that data lives in a high-dimensional vector space; however, many applications involve non-Euclidean data, such as graphs, strings and matrices, or data whose structure is determined by symmetries in the underlying system. Here, we discuss methods for identifying geometric structure in data and how leveraging data geometry can give rise to efficient ML algorithms with provable guarantees.

February 20-21, 2024
New Orleans, Louisiana

Proceedings of the
38th AAAI Conference
on Artificial Intelligence



Edited by Jennifer Dy, Sriram Natarajan & Michael Woodridge

AAAI

PDF

Published

2024-03-24

Effects of Data Geometry in Early Deep Learning

Saket Tiwari
Department of Computer Science
Brown University
Providence, RI 02906
saket_tiwari@brown.edu

George Konidaris
Department of Computer Science
Brown University
Providence, RI 02906

目次

- 幾何学的深層学習
- 幾何学的な表現学習1: 演算
- 幾何学的な表現学習2: 類似度
- データ多様体上の距離

畳み込みニューラルネットワーク

深層学習ブームの火付け役 [Krizhevsky et al., NeurIPS, 2012]

■ 畳み込み演算で実装され、画像認識で高い精度

- 線形演算を畳み込みに制約

■ 画像 x に対しフィルタ w を畳み込むと

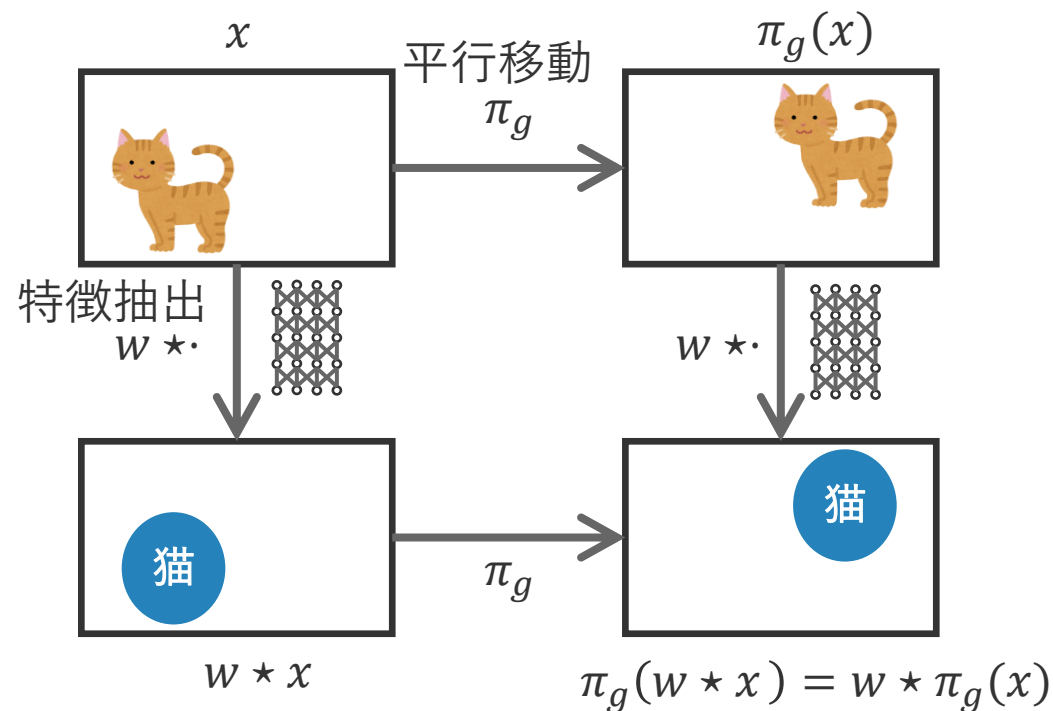
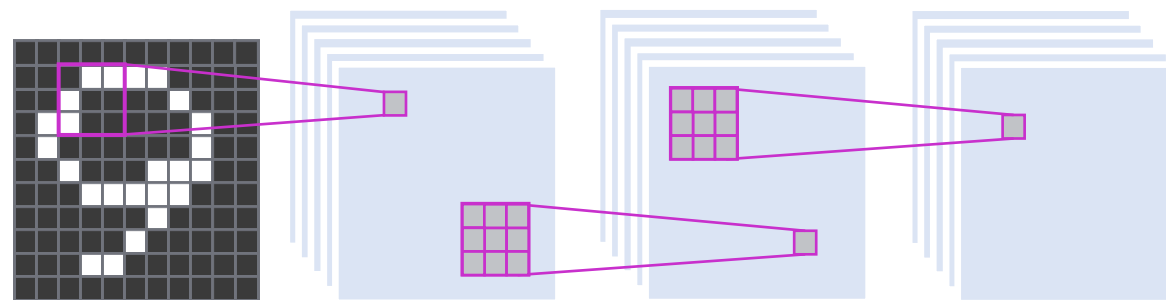
$$(w \star x)[s] = \sum_{u \in \mathbb{Z}^2} w[u] x[u + s]$$

- $\star$ は相互相関演算子
- $x[s]$ は画像 x の位置 $s \in \mathbb{Z}^2$ のピクセル
- 出力は特徴マップ

■ 移動してから畳み込む = 畳み込んでから移動する

- 物体の移動を π_g で表す

$$\begin{aligned} (w \star \pi_g(x))[s] &= \sum_{u \in \mathbb{Z}^2} w[u] x[u - g + s] \\ &= \sum_{u \in \mathbb{Z}^2} w[u - (s - g)] x[u] \\ &= \pi_g(w \star x)[s] \end{aligned}$$



畳み込みニューラルネットワーク

物体の意味はその位置に依存しない

■ 画像中の物体の位置の移動を考える

- 移動量の集合 $g \in G \simeq \mathbb{Z}^2$
- 移動量の演算(並進群) $g + h, -g \in G \ (\forall g, h \in G)$
- 実際に移動させる作用(群作用) $\pi_g, \pi_g(x)[s] = x[s - g]$
- 作用同士の演算 $\pi_{g+h} = \pi_h \circ \pi_g, \pi_{-g} = \pi_g^{-1}$

■ 移動してから畳み込む = 畳み込んでから移動する

- 並進群の構造を保つ = 並進同変

$$w \star \pi_g(x) = \pi_g(w \star x)$$

- プーリングは不変

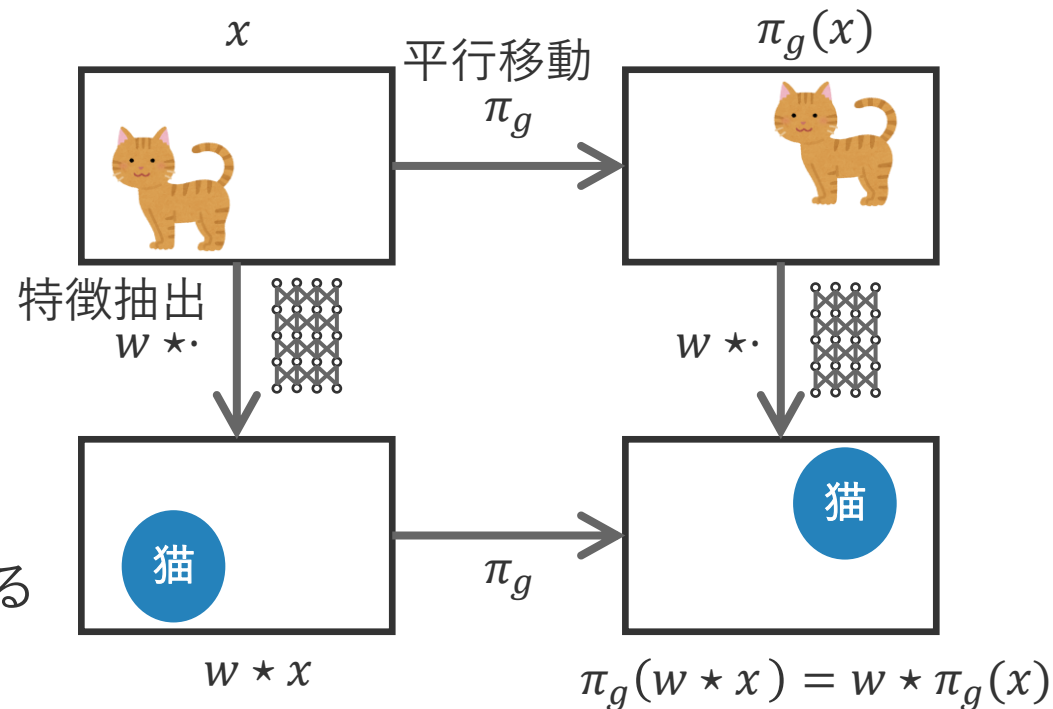
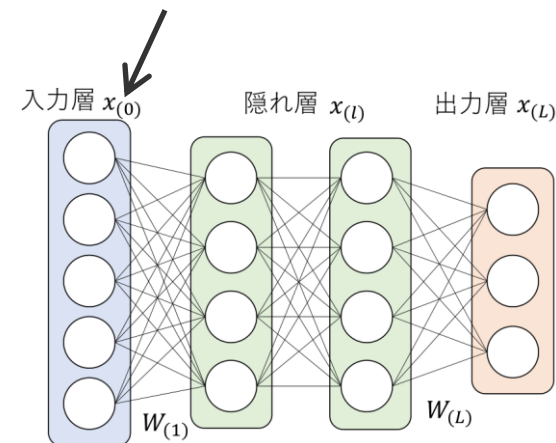
$$POOL(\pi_g(x)) = POOL(x)$$

■ 「違う場所に写った同じ物体」が外挿でなく内挿

- 空間を覆うために必要なデータ数が飛躍的に下がる

[Sannai et al., UAI, 2021]

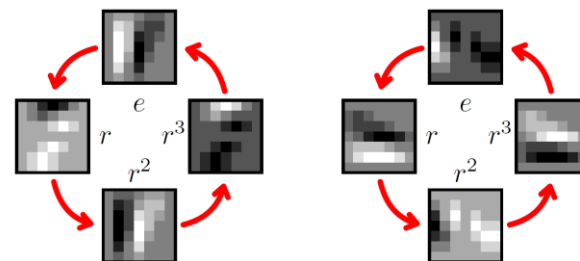
G 不変なら $\mathbb{R}^N$ ではなく $\mathbb{R}^N/G$



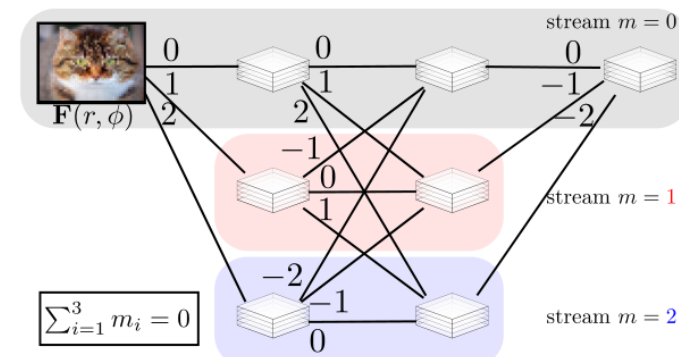
G-畳み込みニューラルネットワーク

画像が持つほかの対称性は？

- ドローンで撮影した画像は向きが不定
⇒ 回転群に対する対称性
- 拡大縮小しても同じ物体
⇒ 拡大縮小群に対する対称性



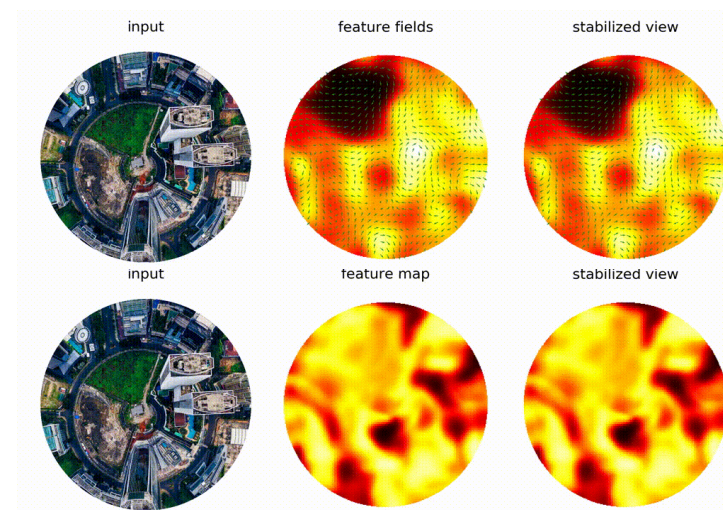
[Cohen et al., ICML, 2016]



[Worrall et al., CVPR, 2017]

拡張する

- 座標を $\mathbb{Z}^2$ から G に拡張する
 - 例：2次の並進群 $\mathbb{Z}^2$ と n 次の離散回転群 C_n を組み合わせる
 - 画像や特徴マップの座標として $s \in G$ を使う
 - 畳み込みの定義はそのまま使える
- 効率的な実装は大変
 - 簡単なのは直積 $\times$ だが冗長
 - 既約(無駄なく理論的)にするには半直積 $\ltimes$



[Weiler&Cesa, NeurIPS, 2019]

[Sosnovik et al., ICLR, 2020]

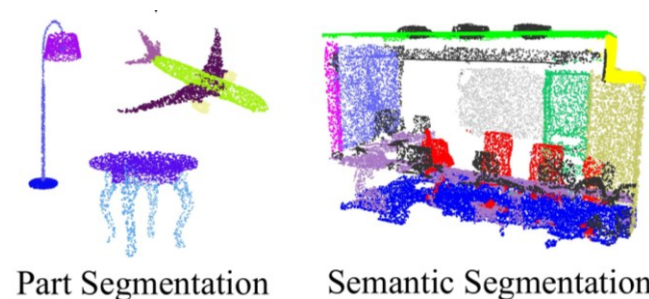
G-畳み込みニューラルネットワーク

順不同なデータ

■ 点群データ

- = 点の順序入れ替え(置換群)に対して対称
⇒ PointNet, DeepSets
- 点ごとに処理して合計や平均を取る
- 分類ならプーリングをして不変に

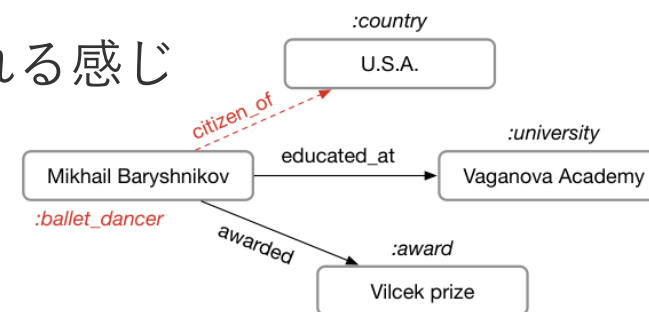
[Qi et al., CVPR, 2017]



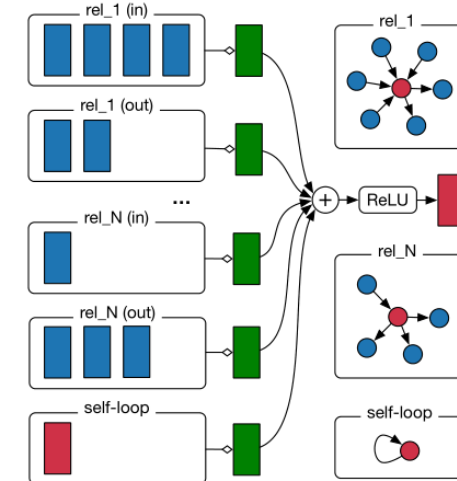
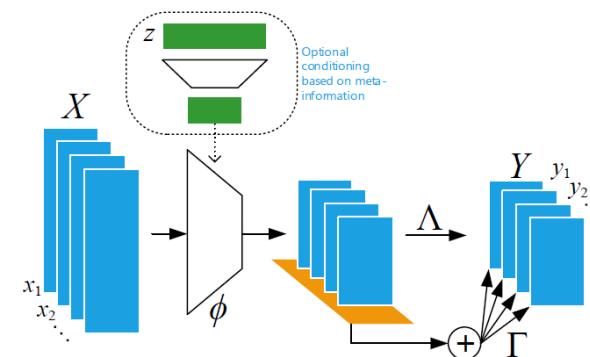
■ グラフ状のデータ

- 頂点やエッジの順序入れ替え(置換群)に対して対称
⇒ グラフニューラルネットワーク
- 入力と出力をエッジで決めてPointNetにイれる感じ

■ 畳み込みの一般化とみなせる



[Bouchacourt et al., AAAI, 2017]
[Matsubara et al., MICCAI, 2018]
[Zaheer et al., NIPS, 2017]



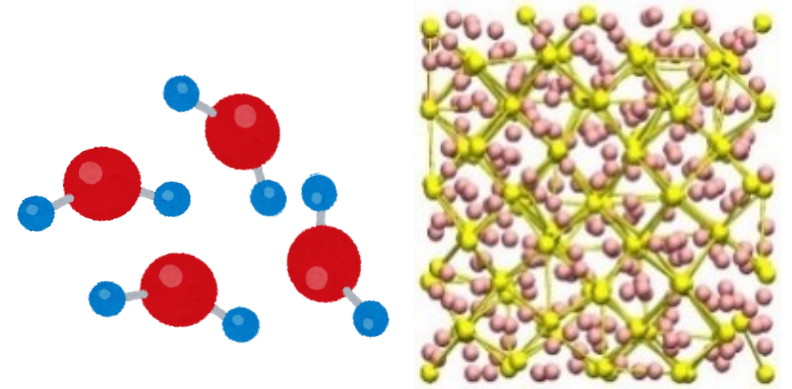
[Schlichtkrull et al., ESWC, 2018]
[Scarselli et al., IEEE TNN, 2009]
[Vaswani et al., NIPS, 2017]

分子シミュレーションのための深層学習

ポテンシャル関数

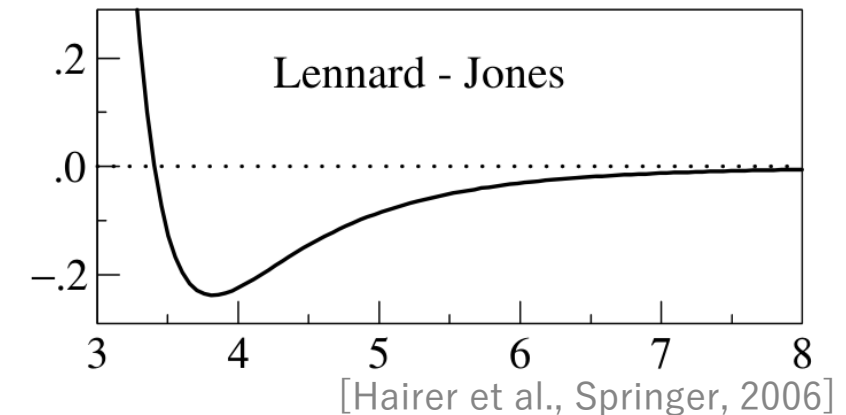
■ 原子間の相互作用(引力・斥力)を計算したい

- 位置エネルギー(ポテンシャル)の勾配の負が力 $m\ddot{q} = -\nabla V(q)$
 - ※重力の例: $m_1\ddot{x}_1 = -\frac{Gm_1m_2}{\|x_1-x_2\|^2} \frac{x_1-x_2}{\|x_1-x_2\|} = -\nabla_{x_1} \left(-\frac{Gm_1m_2}{\|x_1-x_2\|} \right)$
- 原理的には電子状態計算で求められるがあまりに高コスト



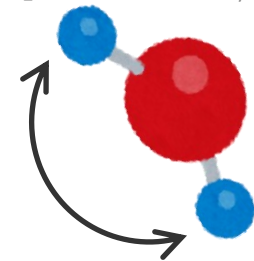
■ 経験的ポテンシャル (20s~)

- (主に) 2 分子間の相互作用のみモデル化
 - 回転・並進不変なので距離にのみ依存
- 計算コストが安いが不正確



■ 機械学習ポテンシャル (90s~)

- 複雑な相互作用をニューラルネットワークやカーネル法でモデル化
 - 教師データは本気で解いたシミュレーション結果
- 3 体以上の関係を取りたいなら, 原点からの距離だけでなく角度も大事



分子シミュレーションのための深層学習

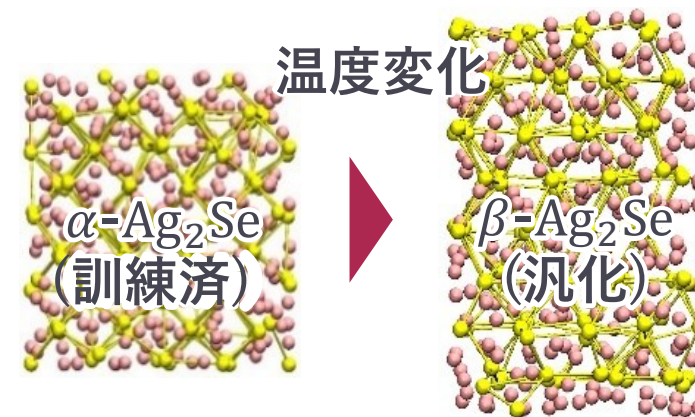
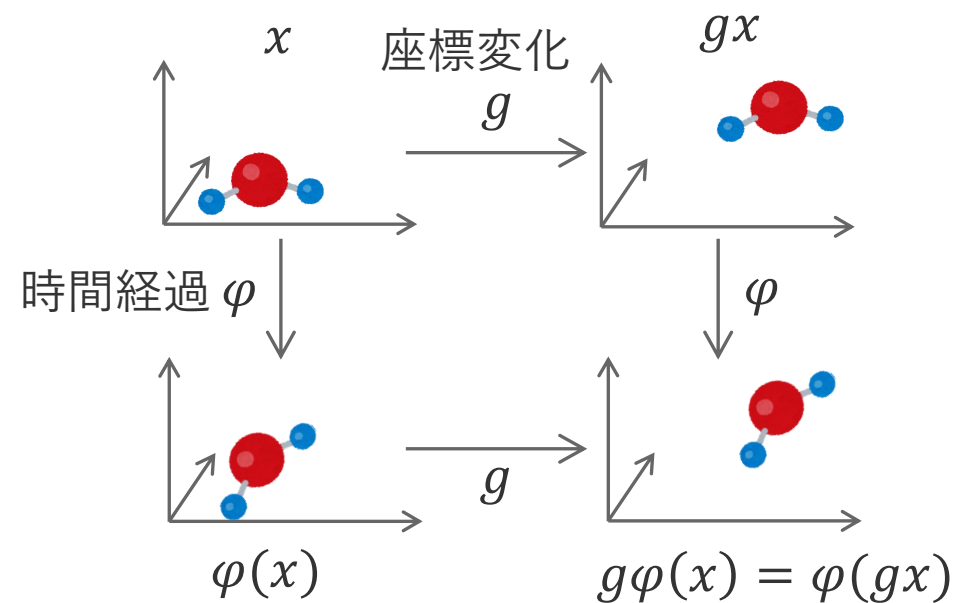
3 分子を相対関係だけで取り出す

■ 目的の原子を原点にする

- 並進不変になるが，回転同変でない
- あらゆる回転パターンを網羅する学習コストが発生

■ 座標の正規化(canonicalization)

- 近い原子や重要な結合の数個を基準に角度を決める
→ 回転同変性が出る
- 専門家の知識に依存，原子の位置に感度が高すぎる
(ちなみに画像の回転同変でもよく使われる)

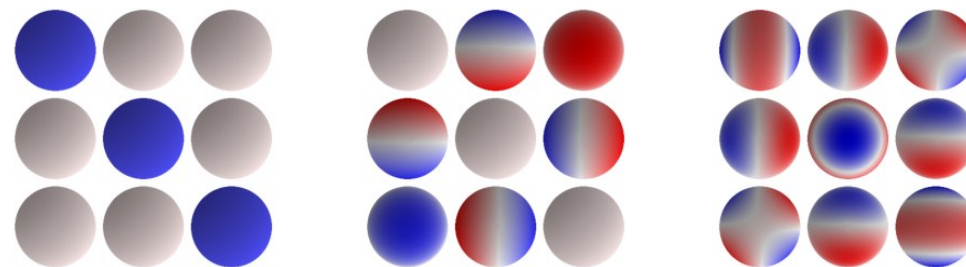
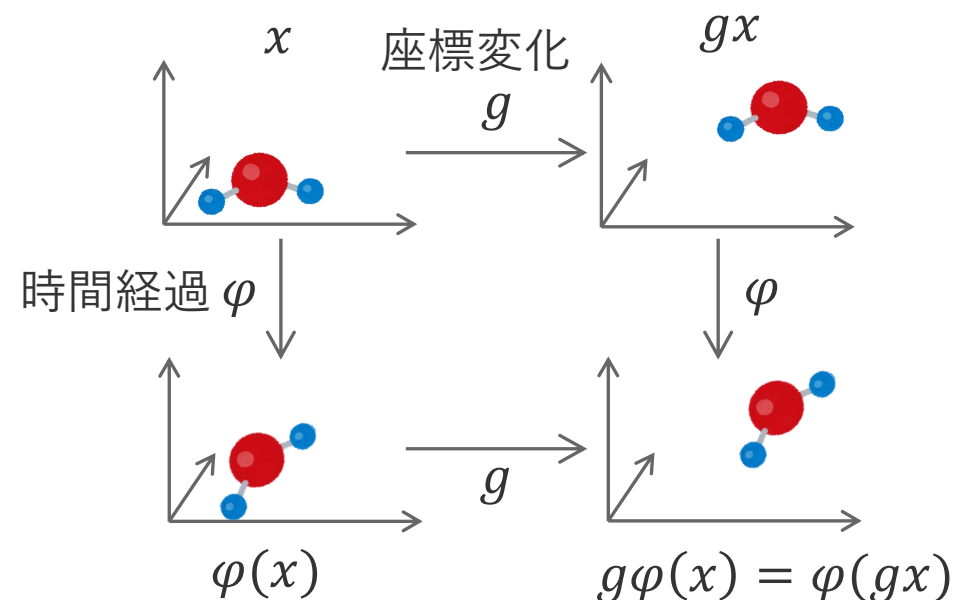


[Shimamura, ..., Matsubara & Tanaka., J. Chem. Phys., 2019]

G-同変ニューラルネットワーク

幾何学的深層学習で深層学習と密結合させる

- 画像は離散化グリッド上のスカラー特徴量の対称性, 分子はベクトル特徴量の対称性(座標・速度)
→ 格子状の畳み込みではない
- 球面調和関数を基底として使う
 - 3次元回転 $SO(3)$ の表現を分解するためのフーリエ基底
 - 複素数で例えると $w_1(w_2z) = w_2(w_1z)$
- クリフォード代数(幾何代数)を使う例も
 - こっちは内積や外積のイメージ
- 抽象的に書くと畳み込みと統一できる
= 幾何学的深層学習
 - 結局は全結合層の制約になっている
 - $E(3)$ on $\mathbb{R}^3 \times \mathbb{R}^3 \times \dots$



[Weiler et al., NeurIPS, 2018]

幾何学的深層学習

幾何学的深層学習 [Bronstein et al., arXiv, 2021]

■ データの対称性を保存する深層学習

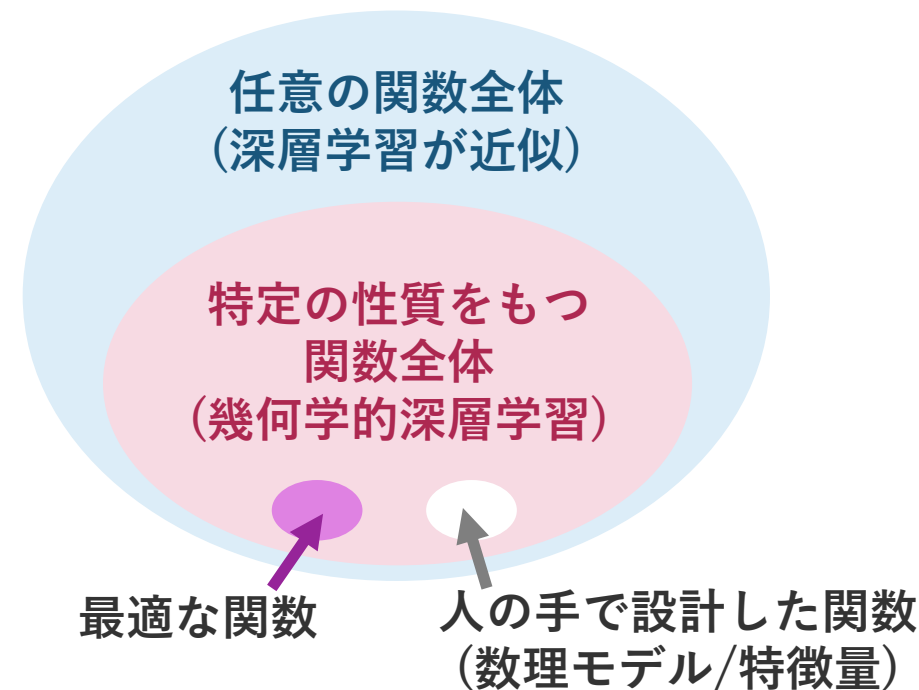
- 特徴量の並び(index)の対称性 ($\mathbb{Z}^n$ on $\mathbb{Z}^n$, S_n on $\{1, \dots, n\}$)
⇒ G-畳み込みニューラルネットワーク
- 特徴量の値の対称性 ($E(3)$ on $\mathbb{R}^3 \times \mathbb{R}^3 \times \dots$)
⇒ G-同変ニューラルネットワーク

■ 各演算 $f: X \rightarrow X$ を G -同変に設計

- $\forall x \in X, \forall g \in G, f(\pi_g(x)) = \pi_g(f(x))$
- f はデータ X の持つ対称性を保存する
- 画像の端が欠けるといった実践的な部分は一旦忘れる

■ Take-home message

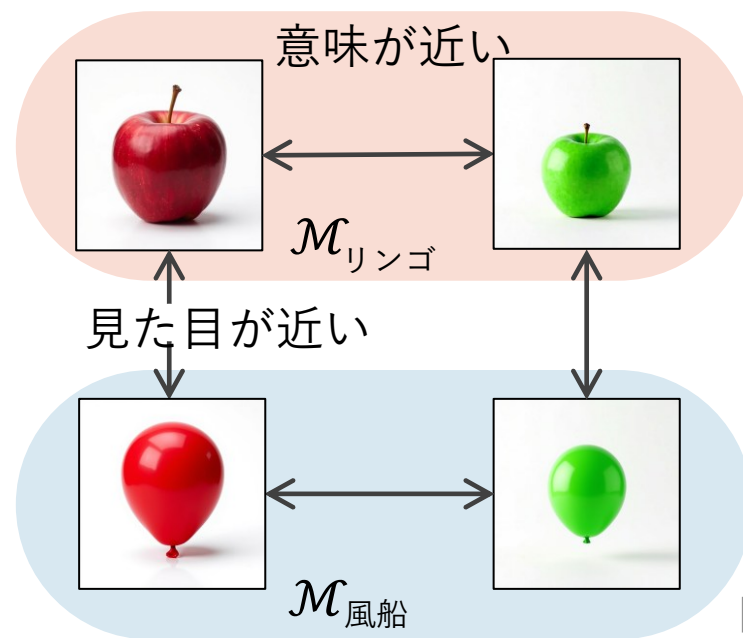
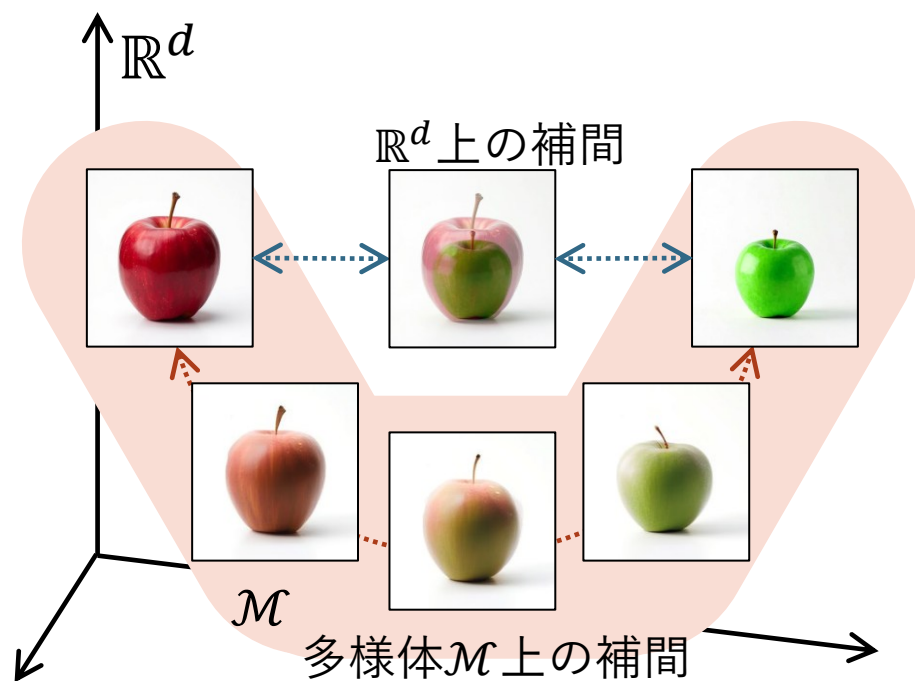
- **たいていの深層学習はデータの幾何学的な構造に合わせて設計されている**
- 設計指針が科学的に与えられる = **Science for AI**
- **では他に活用できるデータの幾何学的な構造はあるか？**



データの幾何学と表現の幾何学

多様体仮説：実データは低次元多様体上にある

- 一般には「連続的に変化する部分集合」くらいの意味
- データ空間で補間すると歪な画像になる
 - データ空間の繋がり $\neq$ 意味的な繋がり
 - データ空間の近さ $\neq$ 意味的な近さ

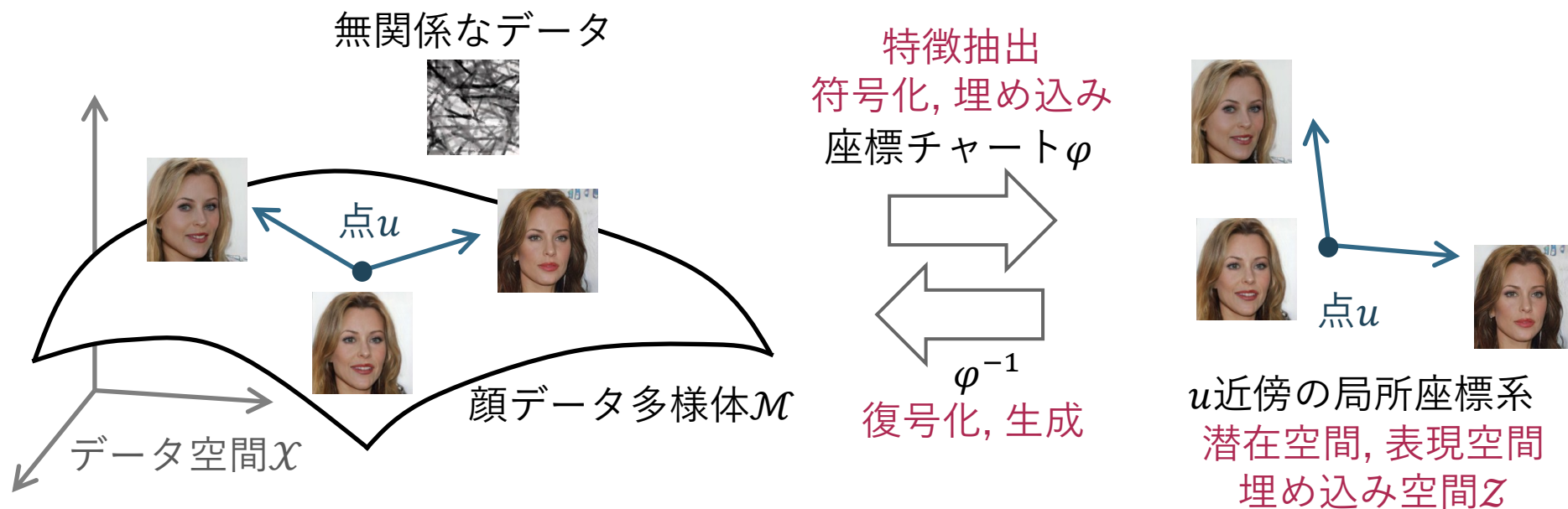


[松原, 数理科学, 2025]

データの幾何学と表現の幾何学

多様体仮説：実データは低次元多様体上にある

- 数学的に多様体であるなら，局所的に低次元ユークリッド空間とみなせる
＝座標チャート φ でデータを低次元に圧縮して処理できる
- データの(低次元)符号化
 - 画像認識→特徴量(feature)，認知科学→表象(representation)
 - 自然言語処理→埋め込み(embedding)，確率モデル→潜在変数(latent variable)
 - 深層学習→中間層(middle layer)，ボトルネック層(bottleneck layer)



データの幾何学と表現の幾何学

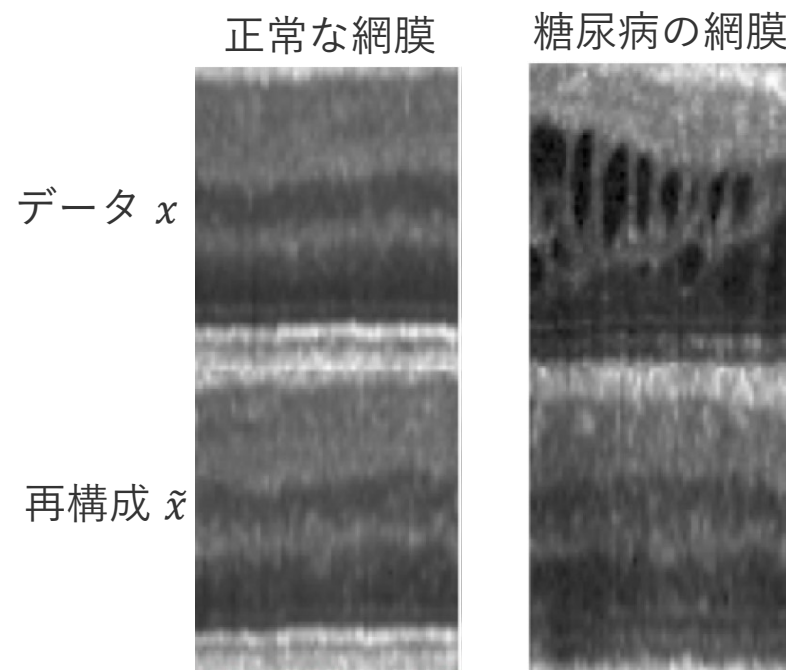
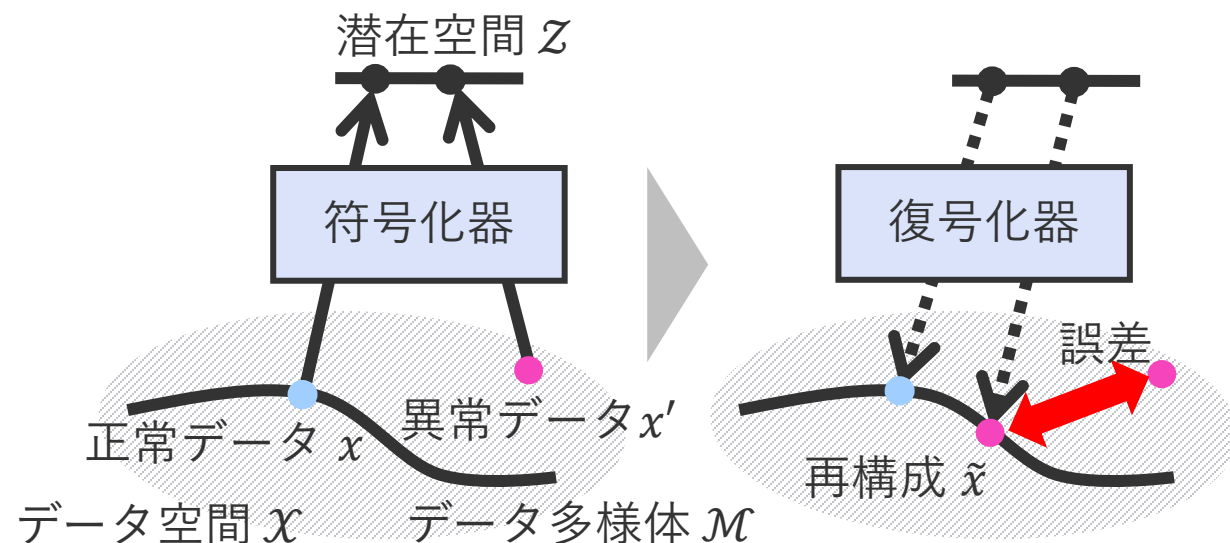
多様体仮説の異常検知への応用

■ 自己符号化器(autoencoder)

- 符号化器 $x \rightarrow z$ と復号化器 $z \rightarrow x$ のペア
- データ多様体 $\mathcal{M}$ 上に無いデータ(異常データ)も $\mathcal{M}$ 上に復元される
= 誤差が生じるので異常と判定できる

■ 敵対的生成ネットワーク(GAN)

- 特殊な方法で復号化器だけ訓練する

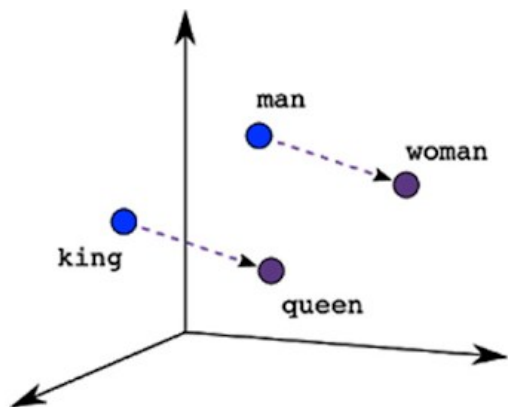


[Schlegl et al., IPMI, 2017]

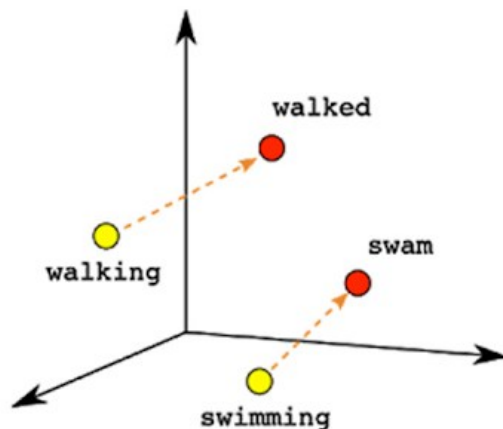
データの幾何学と表現の幾何学

有名な例が単語の埋め込み

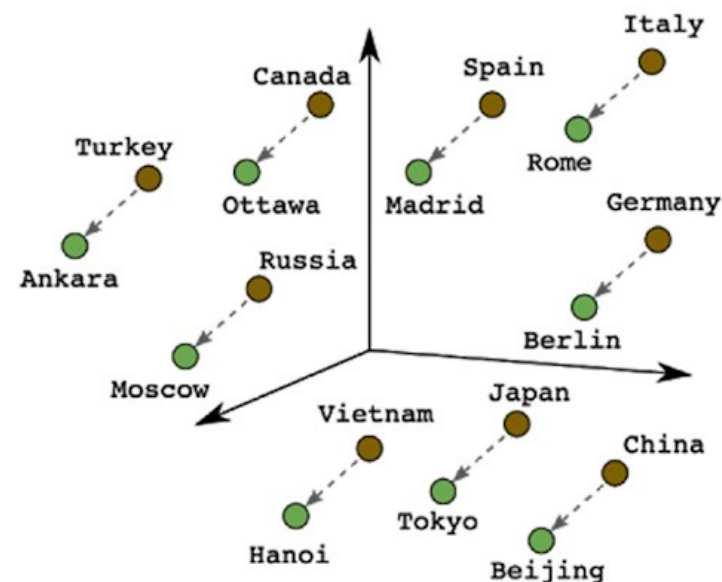
- 単語の埋め込みは、単語→埋め込みベクトル，という写像 ϕ を学習
- 関係性が埋め込みベクトルの差で表現できる
 - 性別ベクトル $\{\text{woman}\} - \{\text{man}\} \simeq \{\text{queen}\} - \{\text{king}\}$
 - 同士の過去形，都市と首都， etc.



Male-Female



Verb Tense



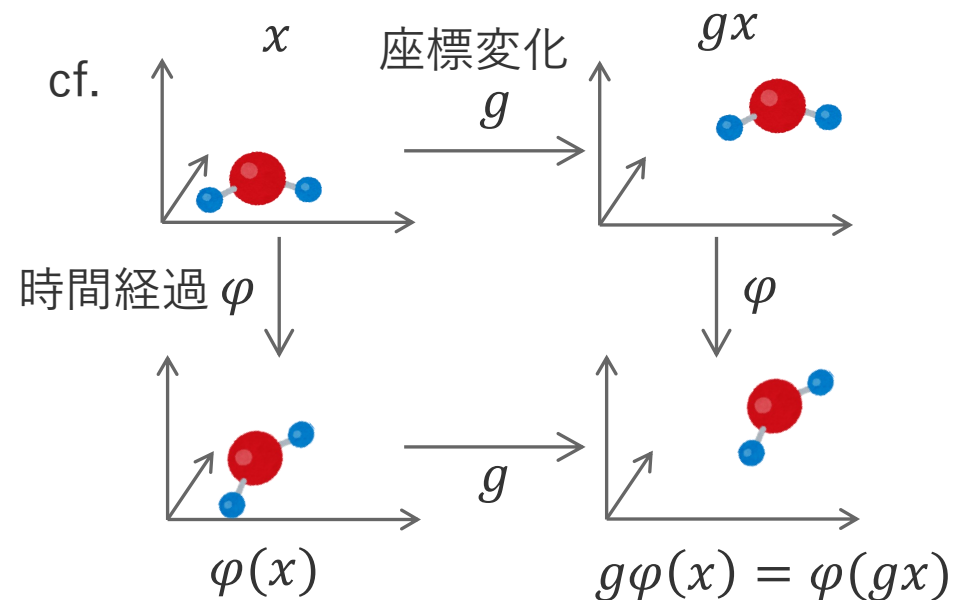
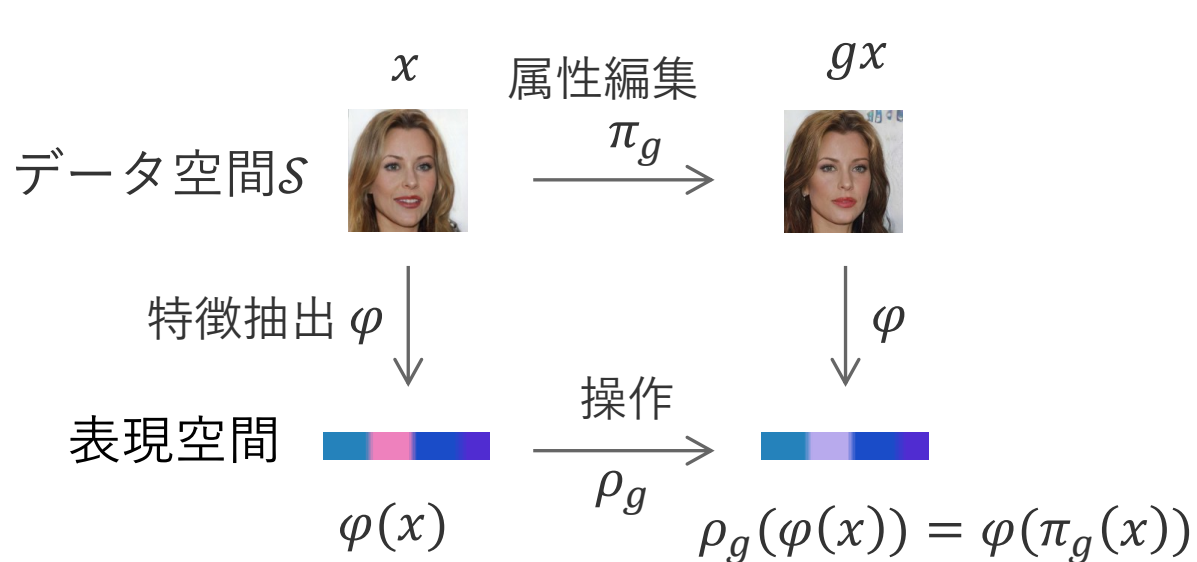
Country-Capital

<https://cloud.google.com/blog/ja/products/ai-machine-learning/example-based-explanations-to-build-better-aiml-models>

データの幾何学と表現の幾何学

画像でも同じことをしたい

- 特徴抽出 φ はデータの操作 π_g を保存して欲しい
 - 表現空間上の操作 ρ_g について, $\rho_g(\varphi(x)) = \varphi(\pi_g(x))$
 - 一般に $\mathcal{Z}$ は数ベクトル空間で, ρ_g はベクトル加算
- 生成側 φ^{-1} でも構わない(例: generative adversarial networks)



表現学習

どのような特徴抽出が適切か？

■ データへの操作同士が独立である必要がある

- 例：表情や向きに依存せずに同一人物判定をしたい

- 操作：「表情を変える」「顔の向きを変える」
- 不変：それが誰であるか，操作対象外の属性(ex. 髪の毛の長さ)

⇒ 1 属性が 1 変数(特徴量) or 1 つの線形独立な数ベクトルに対応

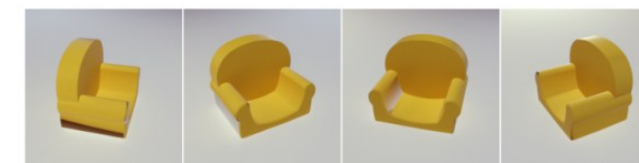
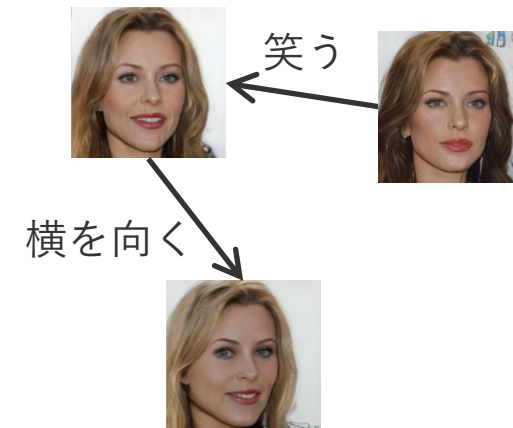
■ もっと一般化して考えると

- 1 属性 = 1 つの群

- 笑い = 加法群($\mathbb{R}, +$) (ただし，上限・下限がありそう)
- 顔の向き = 回転群($S^1, \cdot$) (ただし，後ろ向きは考えなくて良いかも)

- 複数の属性 = 群の直積

- 特徴空間が $\mathbb{R}^N$ ではなく $\mathbb{R} \times S^1 \times \dots$
- 笑顔 $\pi_a: \mathbb{R} \rightarrow \mathbb{R}$ ，顔の向き $\rho_\theta: S^1 \rightarrow S^1$ ，...
- $\pi_a \circ \rho_\theta = \rho_\theta \circ \pi_a$



[Koyama et al., ICLR, 2024] 回転

幾何学的な表現学習(研究紹介)

[Voynov et al., ICML, 2020]
[Tzelepis et al., ICCV, 2021]

画像生成を考える

■ 生成とは潜在変数(特徴量)からデータ(画像)への写像

- 写像 $x = f(z)$ もしくは確率分布 $p(x|z)$ を学習する
- 点の集まりをぐにゃぐにゃするイメージ

■ 潜在変数 z は適当な分布 $p(z)$ に従う

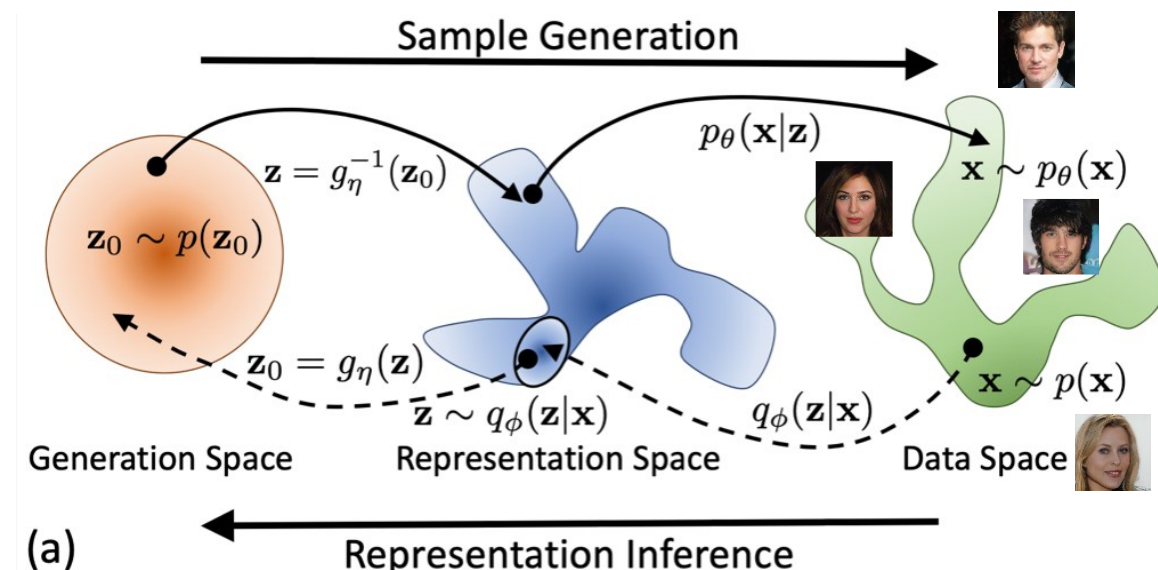
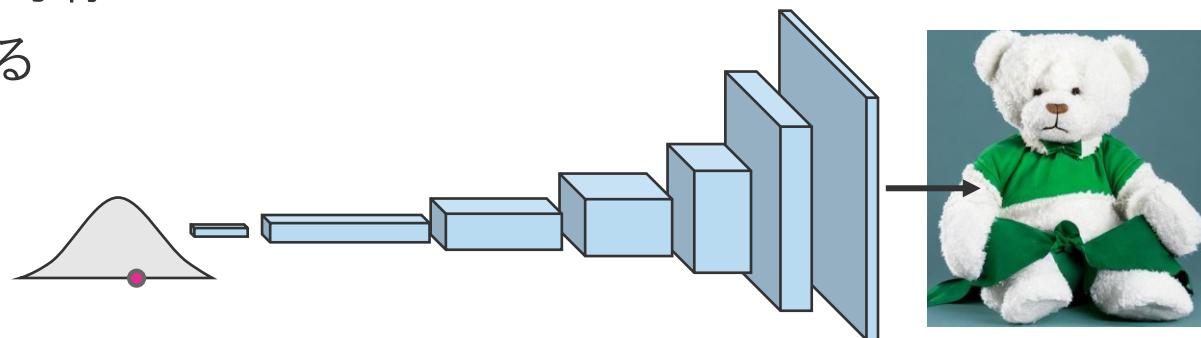
- たいてい正規分布

■ データ x の分布は

$p(x) = \int_z p(x|z)p(z)$ で定義される

- 典型的なデータほど $p(x)$ は高くあるべき
⇒ リアルなデータのみ生成される

■ 逆写像 $z = g(x)$ は特徴抽出



https://www.sci.utah.edu/~shireen/deep_generative_models.html

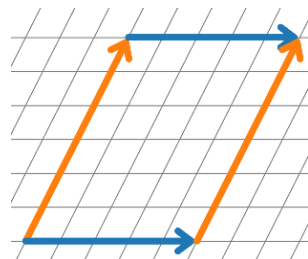
幾何学的な表現学習(研究紹介)

学習済み画像生成モデルを用いた画像編集

■ 潜在変数 z を操作することで生成画像 x を変える = 編集

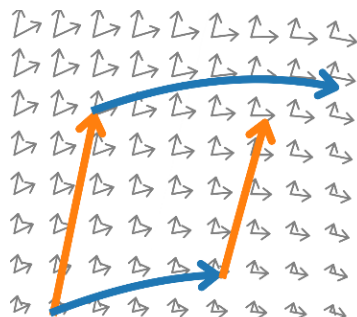
■ Word2vecみたいに線形な操作を想定する

- $(\mathbb{R}, +) \times (\mathbb{R}, +) \times \dots$ に相当
 - $\{\text{king}\} + (\{\text{woman}\} - \{\text{man}\}) \simeq \{\text{queen}\}$
- disentangledだが仮定が強い
(実データの分布は歪んでいる)
 - 学習済みではうまくいかない
 - 学習時に強制すると性能低下

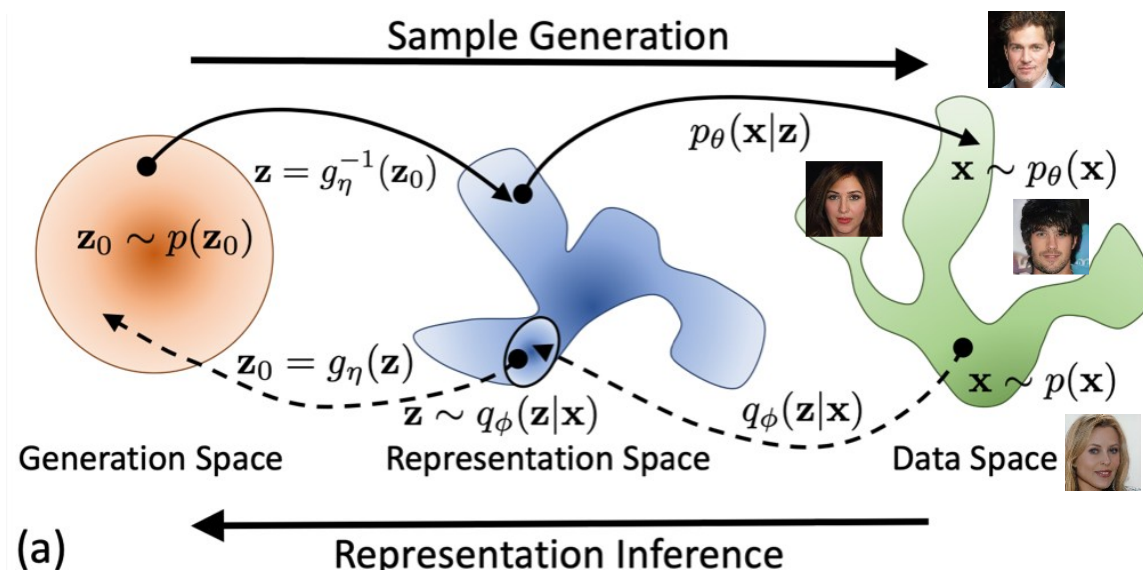
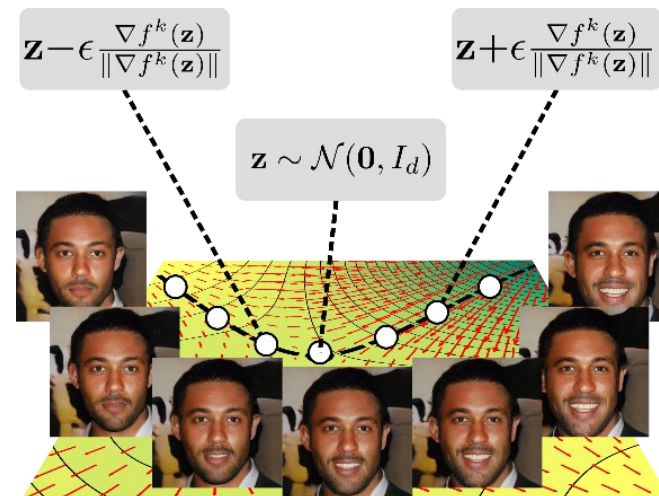


■ ベクトル場を積分する

- 非線形で柔軟
- 非可換である(非disentangled)
 $(\pi_a \circ \pi_b \neq \pi_b \circ \pi_a)$



[Voynov et al., ICML, 2020]
[Tzelepis et al., ICCV, 2021]



https://www.sci.utah.edu/~shireen/deep_generative_models.html

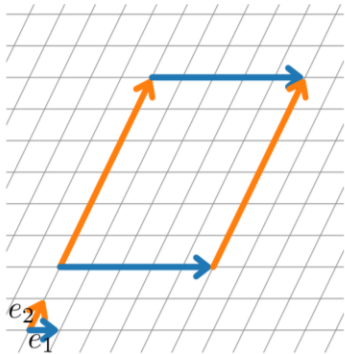
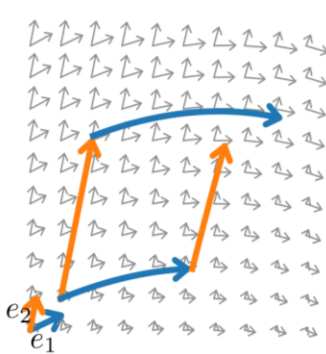
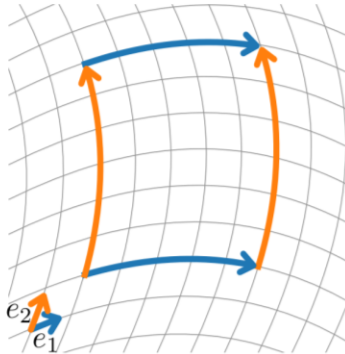
幾何学的な表現学習(研究紹介)

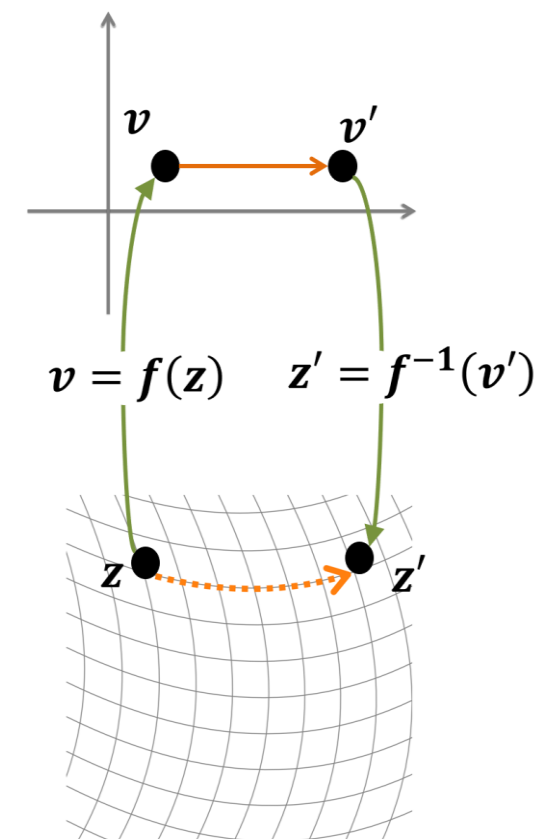
[Aoshima&Matsubara, CVPR, 2023]

学習済み画像生成モデルを用いた画像編集に適した潜在空間の演算は？

- 可換なベクトル場(=曲線座標系)のフローを用いて編集を定義
 - ベクトル空間の構造(disentangled)を保ちながら非線形化

	Linear arithmetic	Vector fields/Local basis	DeCurvEd
Global coordinate	oblique	(only local)	curvilinear
No retraining	✓	✓	✓
Nonlinear edit	✗	✓	✓
Commutative edit	✓	✗	✓

Conceptual diagram			
--------------------	--	--	--



幾何学的な表現学習(研究紹介)

[Aoshima&Matsubara, CVPR, 2023]

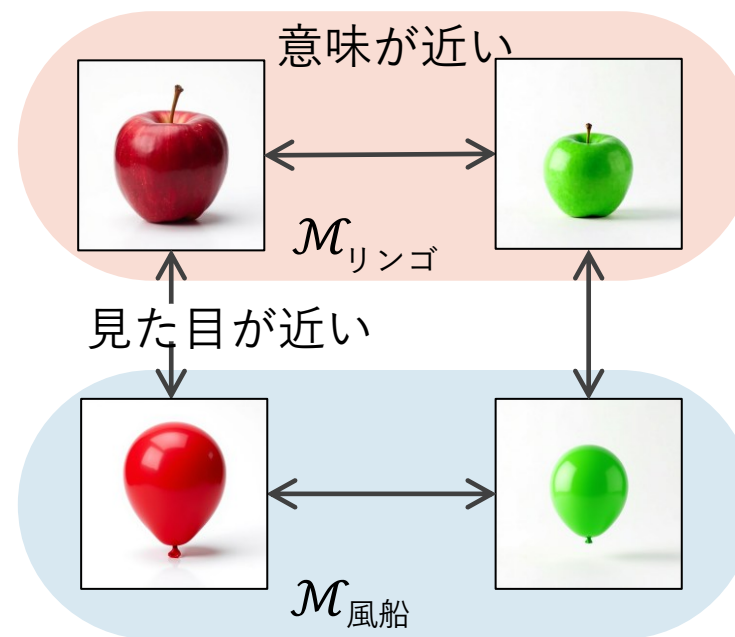
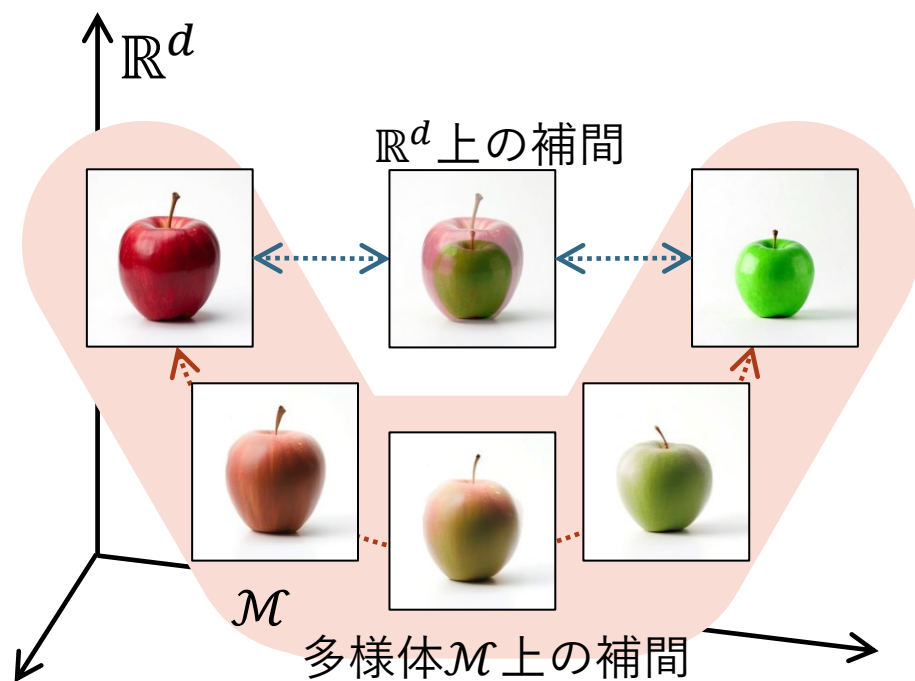
精度(柔軟性)と可換性(対称性)を両立



データ多様体と類似度

多様体仮説を使ってデータ間の類似度を定めたい

- 演算の保存はかなり制約が大きい
- せめて類似性だけでも保存したい



[松原, 数理科学, 2025]

埋め込みの幾何学

マルチモーダル埋め込みで明示的に“近さ”を学習する

■ 類似度(近さ)の尺度を定義する

- 対応する画像とテキストのペアを用意
- 正ペアの類似度を高く，負ペアの類似度を低く訓練(対照学習)

⇒データ点間の類似度が学習される

■ 検索やfew-shot分類が可能

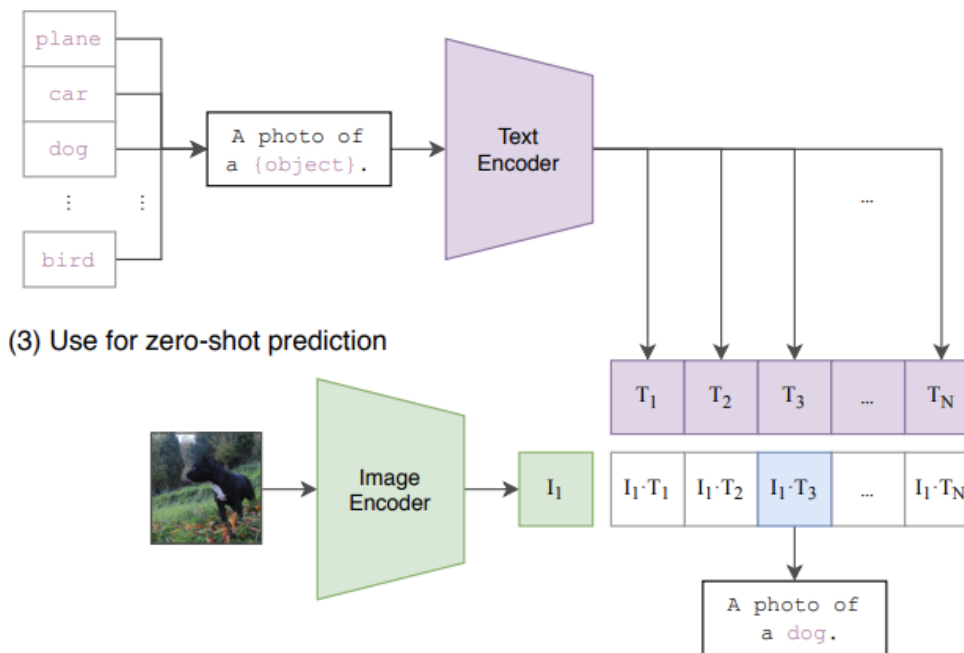
- クエリに対して一番類似したものを返す
- 音声検索，推薦システムなども

■ 類似度の定義は様々

- 距離の近さ，内積の大きさ，コサイン類似度， etc
- 距離と内積は非有界なので，コサイン類似度が人気
 - ex. CLIP
 - コサイン類似度=長さ 1 の内積 $\frac{\langle x_1, x_2 \rangle}{\|x_1\| \|x_2\|}$
 - 球面上で近い

画像からテキストの検索(≒分類)

(2) Create dataset classifier from label text



(3) Use for zero-shot prediction

[Radford et al., ICML, 2021]

埋め込みの幾何学

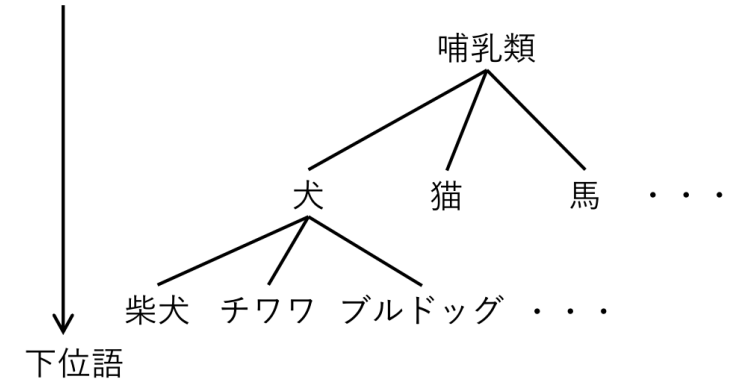
単語は階層性を持つ

■ 単語の階層性は(ほぼ)木構造

- 例：
 - 哺乳類 ⊃ 犬, 猫, 馬
 - 犬 ⊃ 柴犬, チワワ, ブルドック
- 検索の妥当性は(ほぼ)木の上の距離
 - 「犬」と検索して「柴犬」が出てくるのはOK (距離 1)
 - 「チワワ」と検索して「柴犬」が出てくるのはNG (距離 2)

■ 子要素が指数的に増える

- 完全 n 分木では, 深さに対して幅が指数的 $\#leaves \propto n^{\#depth}$:
 - ユークリッド空間では, 原点からの距離に対して表面積が多項式 (d 次元超球の表面積 $\propto r^{d-1}$)
 - ユークリッド空間や球面上では狭すぎる
- ※表現できないわけではない



埋め込みの幾何学

木構造は双曲的

■ 双曲的な空間とは？

- Gromovの意味での双曲性
 - 3点間を結ぶ線が互いに近い
 - 典型例が双曲空間と木構造(metric tree)
- 双曲空間とは曲率が負で一定の空間
 - ユークリッド空間中で表現すると鞍馬型

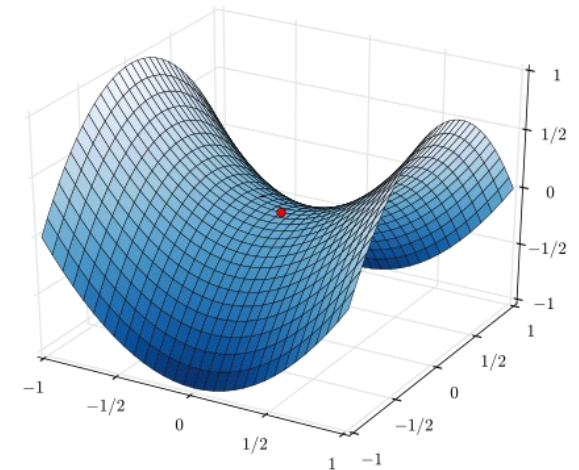
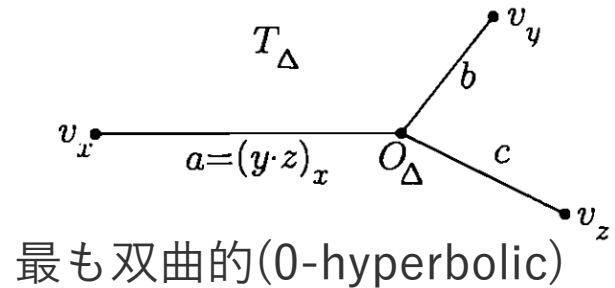
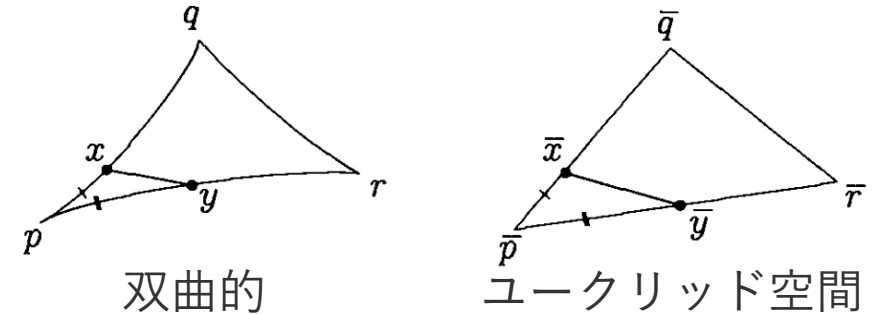
■ Sarkar (2011)

- 妥当な条件下で, metric tree $\rightarrow$ 2次元双曲空間という写像で, “歪み”(距離の誤差)が定数倍で抑えられるものがある
- $\Rightarrow$ 木構造は双曲空間への埋め込みできれいに表現できる

[Sarkar, International Symposium on Graph Drawing, 2011]

[Nickel&Kiela, NIPS, 2017]

[Bridson&Haefliger, Springer, 1999]

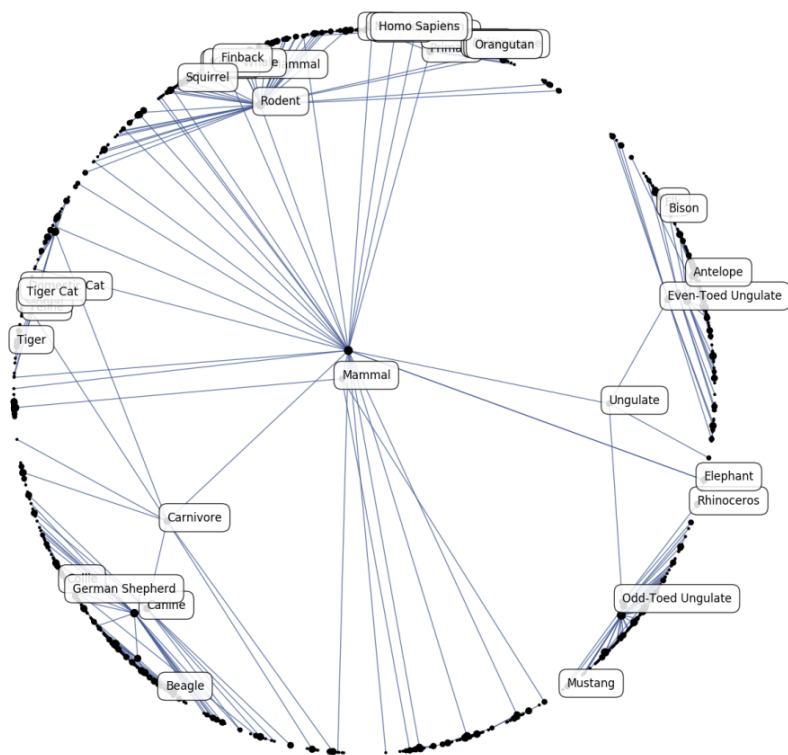


https://en.wikipedia.org/wiki/Saddle_point

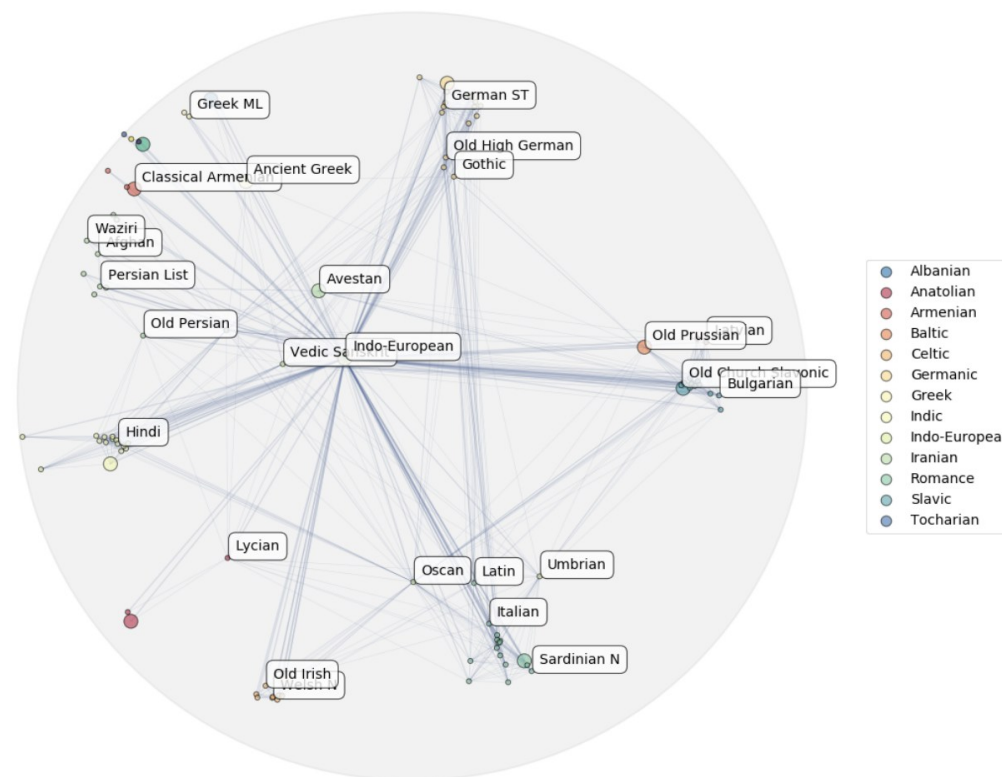
埋め込みの幾何学

階層性には双曲空間

- 単語以外にも系統樹(言語, 動物種など)も



[Nickel&Kiehl, NIPS, 2017]



[Nickel&Kiehl, ICML, 2018]

埋め込みの幾何学

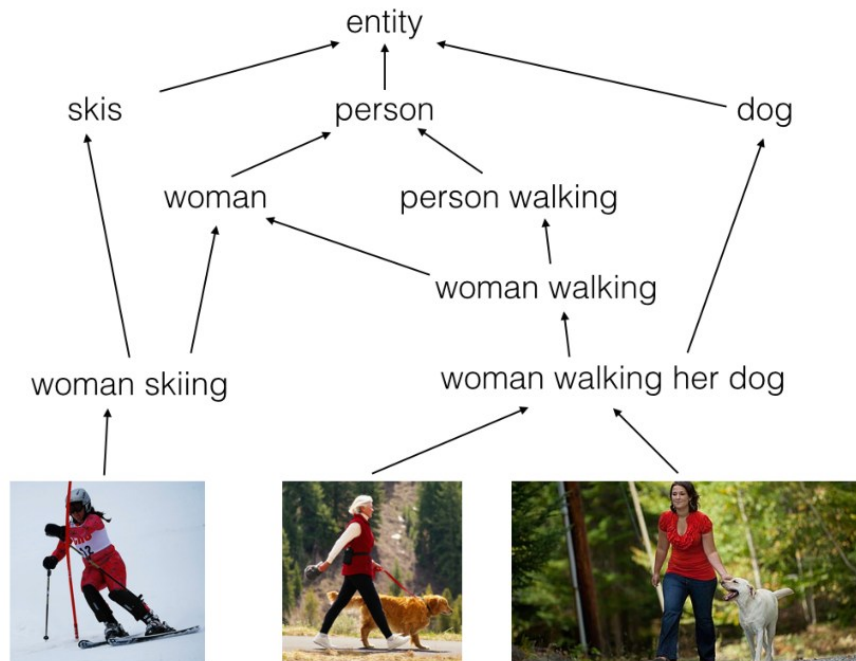
マルチモーダルな階層性

■ 階層構造とは含意(→)

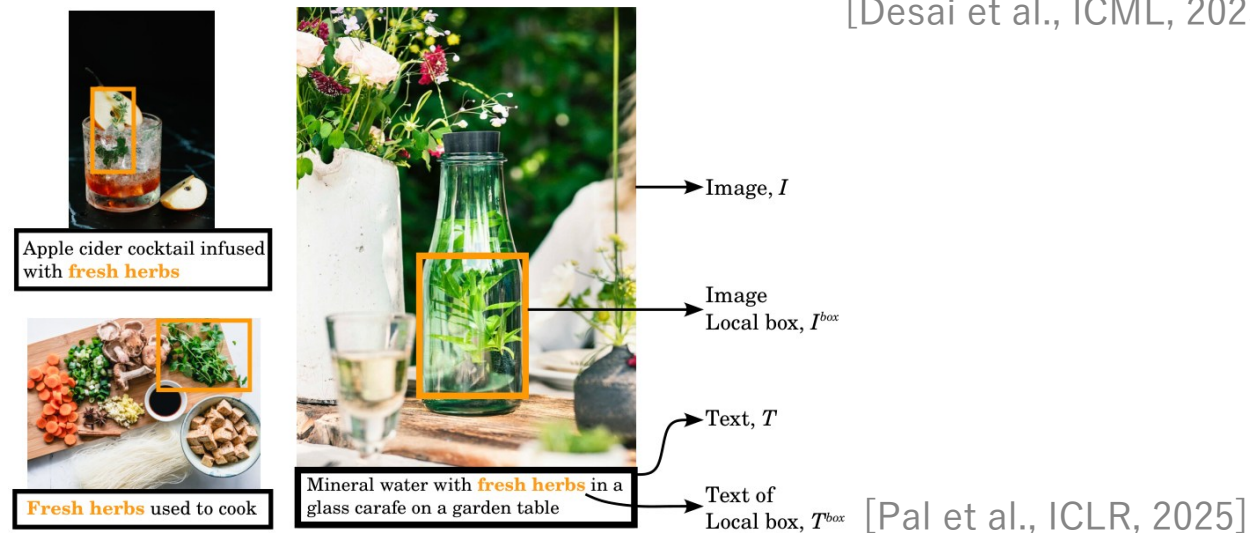
- 子概念→親概念 (ex. 犬→哺乳類)
- 具体的な文章→曖昧な文章
- 画像の全体→画像の部分 (共起→単独)
- 文章の全体→文章の部分
- 画像→文章

■ CLIPでも双曲空間への埋め込みが有効

- 必ずしも木構造でなくとも、
子ノードが増えるのは同じ



[Vendrov et al., ICLR, 2016]
[Desai et al., ICML, 2023]



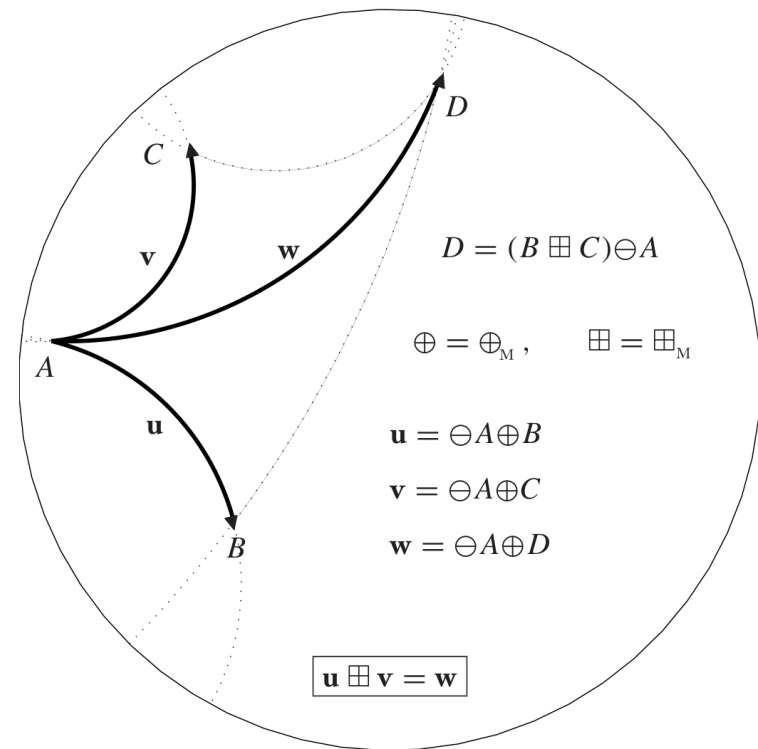
[Pal et al., ICLR, 2025]

埋め込みの幾何学

双曲空間は本当に適切か？

- きれいな演算がない = disentangledにならない
 - 回転と半径への分割のみ可能
 - メビウス和が定義されているが、可換でない(結合的ではない)
 - vs きれいな(disentangled)表現は群の直積(可換性を持つ)

[Higgins et al., arXiv:1812.02230]



[Ungar, Springer, 2009]

埋め込みの幾何と代数

[Bhuyan et al., Neural Computing and Applications, 2024]

階層構造と演算を両立させる

■ 階層性は要素の包含関係

- 犬 = {うちの犬, ハチ公, 太郎, ...}
- 犬 → 哺乳類 というより 哺乳類 ⊃ 犬
- もしくは $\forall x (x \in \text{犬} \rightarrow x \in \text{哺乳類})$
- これを概念の外延的定義という

■ 階層性は属性の包含関係(逆向き)

- 犬 = $\{x | x \text{は体毛, 四本脚, 肉食, etc}\}$
- 哺乳類 = $\{x | x \text{は体毛, 四本脚, etc}\}$
- 哺乳類は属性 a を持つ → 犬は属性 a を持つ
- これを概念の内包的定義という

■ 一致すると嬉しいが一般にはそうではない

[Ganter&Wille, Springer, 1999]

		Localist Representation						Distributed Representation					
		Bear	Tiger	Eagle	Fish	Turtle	Frog	Walk	Swim	Fly	Egg	Claw	Wild
Bear		●						●	●	○	○	●	●
Tiger			●					●	●	○	○	●	●
Eagle				●				○	○	●	●	●	○
Gold Fish					●			○	●	○	●	○	○
Turtle						●		●	●	○	●	○	○
Frog							●	●	○	○	●	●	○

埋め込みの幾何と代数

階層構造と演算を両立させる

■ 順序ベクトル空間を使う order embedding

- 座標の上下関係で表現
 - $y \rightarrow x \Leftrightarrow \forall i. y_i \geq x_i$
 - 座標を属性と見ると内包的
- 右上空間の包含関係ともみなせる
 - $y \rightarrow x \Leftrightarrow R[y] \subset R[x]$
 - 点の集合と見ると外延的

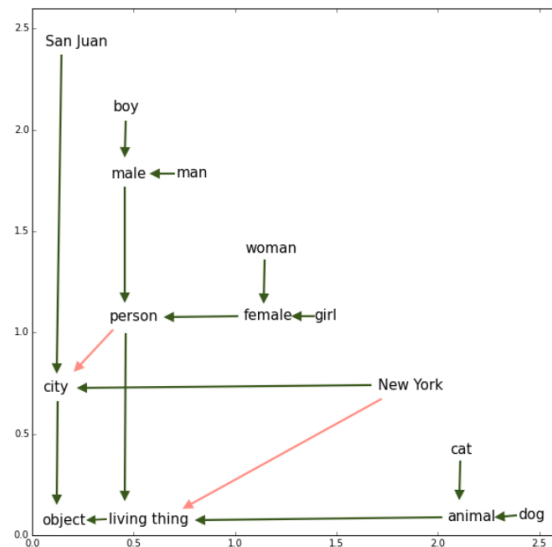
■ 演算ができる

- $\forall i. z_i = \max(x_i, y_i) \Leftrightarrow R[z] = R[y] \cap R[x] \Rightarrow z \rightarrow x, y$
- $\forall i. z_i = \min(x_i, y_i) \Rightarrow R[z] \supset R[y] \cup R[x] \Leftrightarrow x, y \rightarrow z$

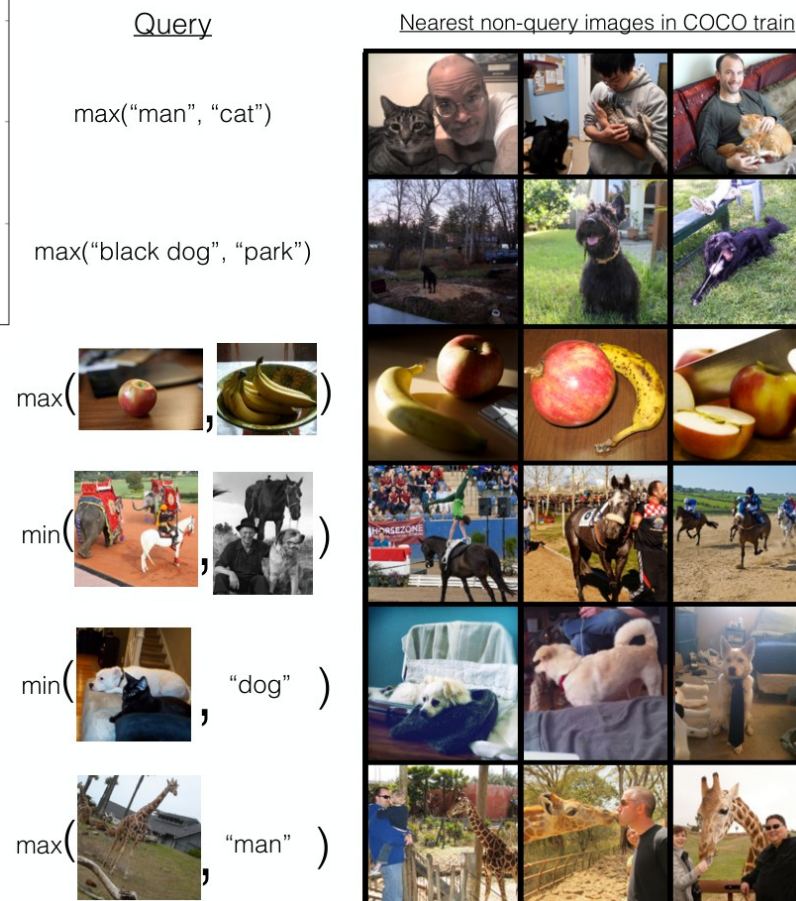
■ 実質的にはブール代数 $\{0,1\}^n$ の $\mathbb{R}^n$ への拡張

- $\max$ = 属性の論理和 $\vee$ = 要素の積集合 $\cap$
- $\min$ = 属性の論理積 $\wedge$ = 要素の和集合 $\cup$

[Yoshikawa&Matsubara, ICLR, 2026]



[Vendrov et al., ICLR, 2016]



埋め込みの幾何と代数

階層構造と演算を両立させる

■ 順序ベクトル空間を使う order embedding

- 座標の上下関係で表現
 - $y \rightarrow x \Leftrightarrow \forall i. y_i \geq x_i$
 - 座標を属性と見ると内包的
- 右上空間の包含関係ともみなせる
 - $y \rightarrow x \Leftrightarrow R[y] \subset R[x]$
 - 点の集合と見ると外延的

■ 演算ができる

- $\forall i. z_i = \max(x_i, y_i) \Leftrightarrow R[z] = R[y] \cap R[x] \Rightarrow z \rightarrow x, y$
- $\forall i. z_i = \min(x_i, y_i) \Rightarrow R[z] \supset R[y] \cup R[x] \Leftrightarrow x, y \rightarrow z$

■ 実質的にはブール代数 $\{0,1\}^n$ の $\mathbb{R}^n$ への拡張

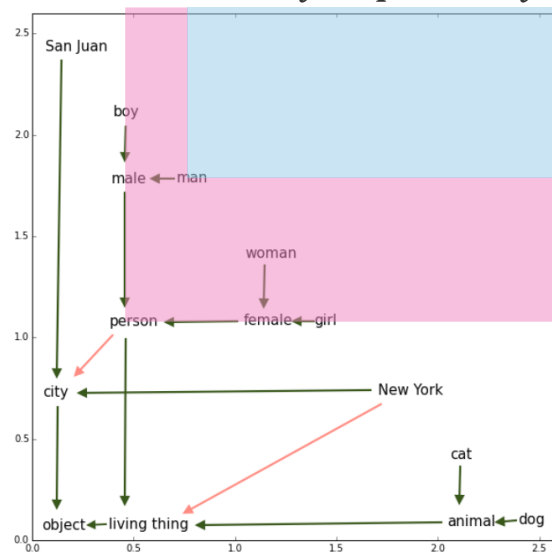
- $\max$ = 属性の論理和 $\vee$ = 要素の積集合 $\cap$
- $\min$ = 属性の論理積 $\wedge$ = 要素の和集合 $\cup$

[Yoshikawa&Matsubara, ICLR, 2026]

2026.7.23

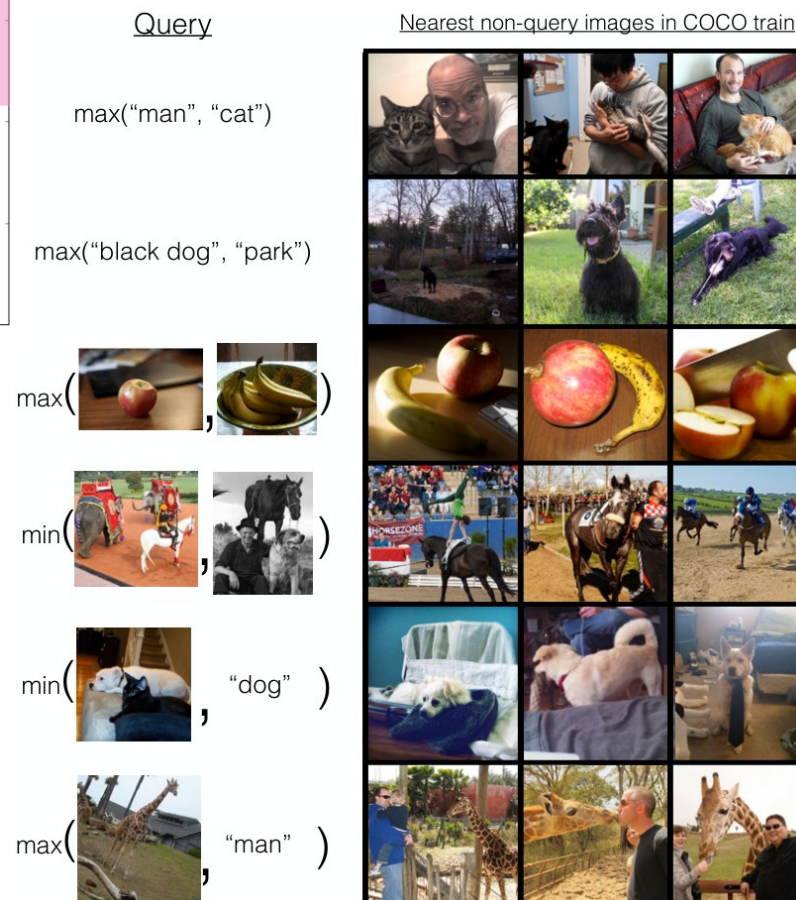
「学習物理学の創成」領域セミナー

man $\subset$ person
man $\rightarrow$ person
man_i $>$ person_i



male $\cup$ female = person
min(male_i, female_i) = person_i

[Vendrov et al., ICLR, 2016]



埋め込みの幾何と代数

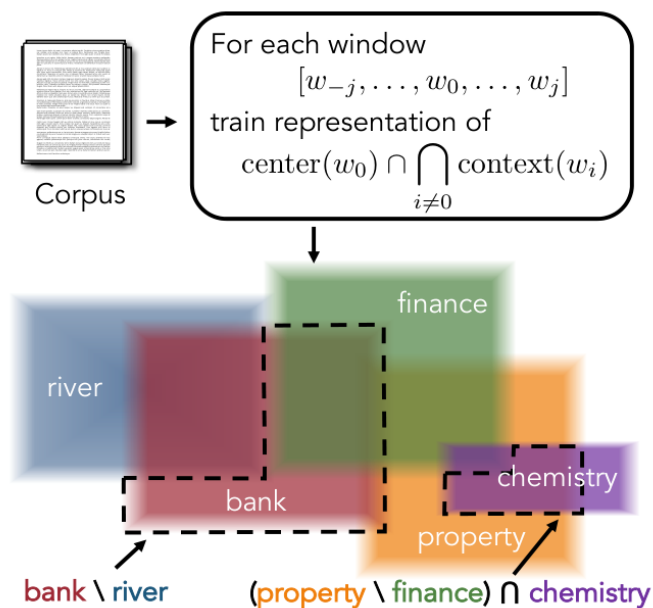
領域の包含関係で表現した例

■ 演算が閉じない

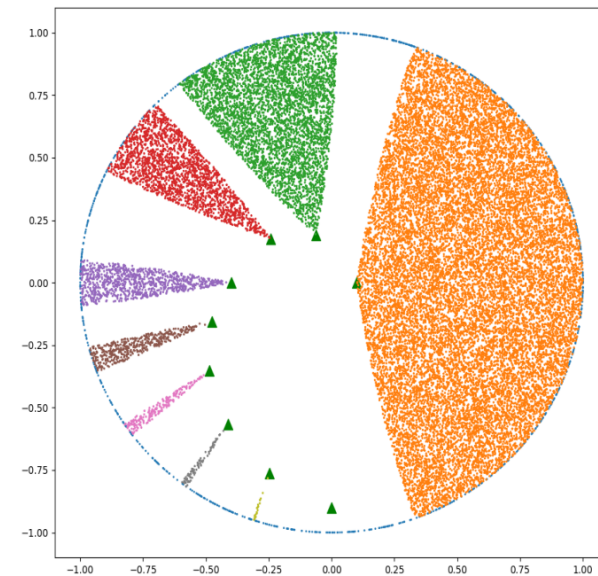
- 原点から外向きの三角錐(entailment cone)
 - 双曲空間の木構造と相性が良い
 - 前述の双曲版CLIPでも多用される [Desai et al., ICML, 2023] [Pal et al., ICLR, 2025]
- 超球(disk embedding)

■ 演算が部分的に閉じる

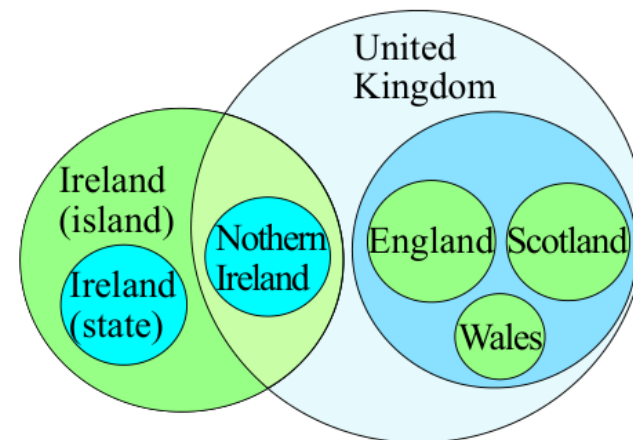
- 超立方体(box embedding)
 - 積集合は閉じるが差集合は駄目



[Dasgupta et al., ACML, 2022]



[Ganea et al., ICML, 2018]



[Suzuki et al., ICML, 2019]

埋め込みの幾何と代数(研究紹介)

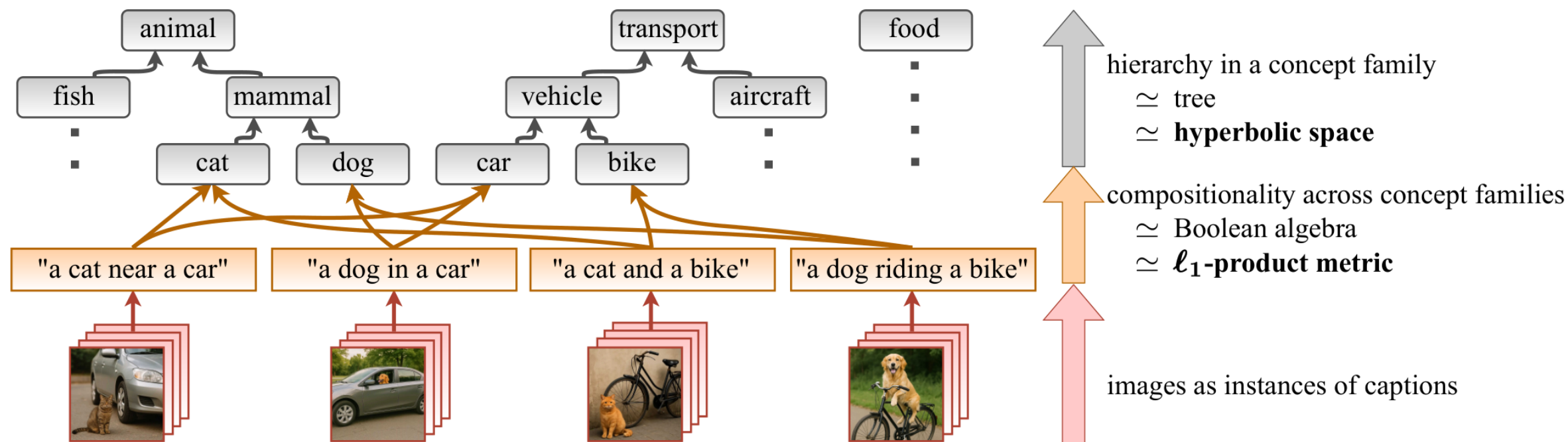
[Yoshikawa&Matsubara, ICLR, 2026]

階層性と演算は両立しないか？

■ 複数の階層性を分けて考える

- 概念の階層は木構造的 = 双曲空間上の距離
- 共起関係による階層はブール代数的 = ハミング距離(ℓ_1 計量)

⇒ 双曲空間のデカルト積に ℓ_1 -直積計量を入れる



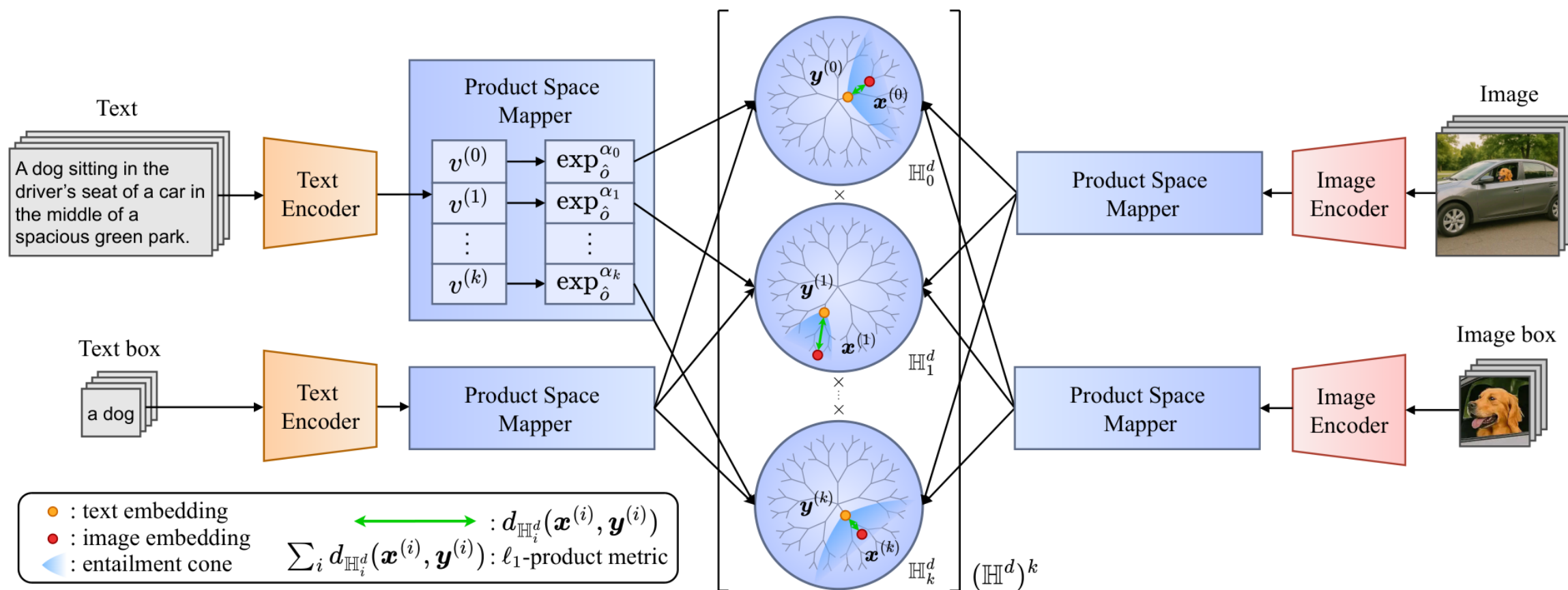
埋め込みの幾何と代数(研究紹介)

[Yoshikawa&Matsubara, ICLR, 2026]

双曲空間のデカルト積に ℓ_1 -直積計量を入れる

■ $d(X, Y) = \sum_i d_{\mathbb{H}_i^d}(\mathbf{x}_i, \mathbf{y}_i)$ for $X = (\mathbf{x}_1, \dots, \mathbf{x}_n), \mathbf{x}_i \in \mathbb{H}_i^d$

- あと entailment cone も使う

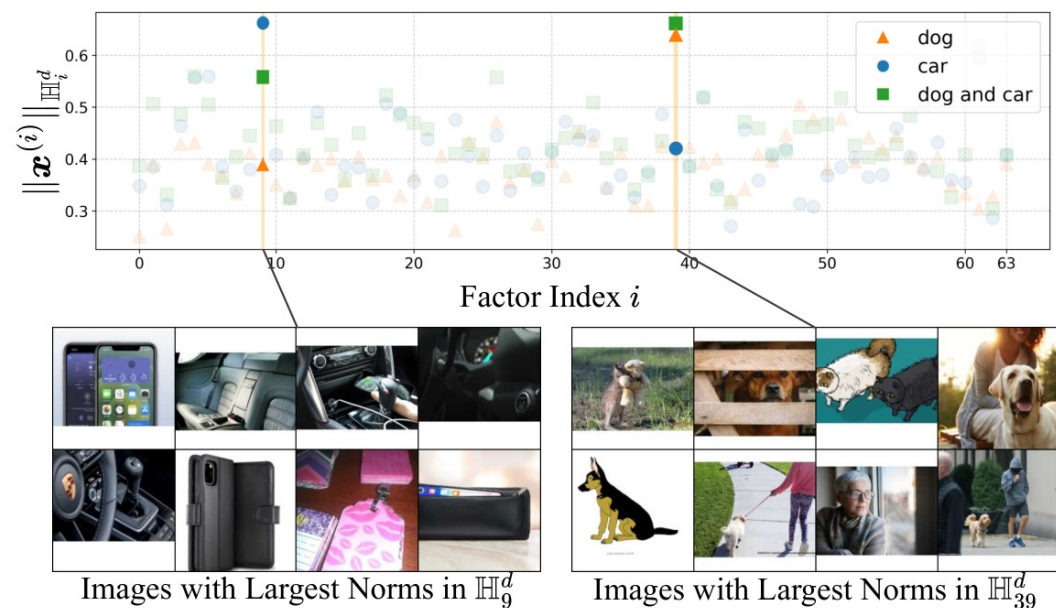


埋め込みの幾何と代数(研究紹介)

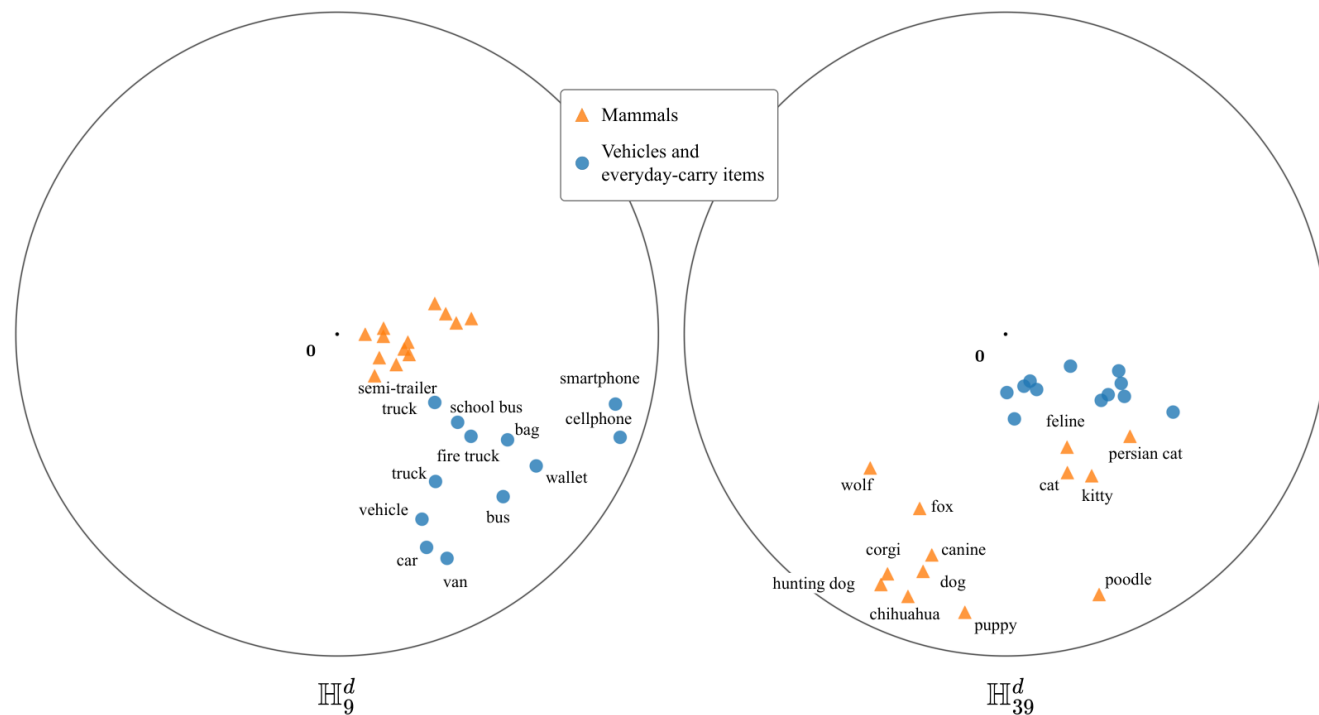
[Yoshikawa&Matsubara, ICLR, 2026]

双曲空間のデカルト積に ℓ_1 -直積計量を入れる

- 1 双曲空間 = 1 概念ファミリーの階層性が創発
- ブール代数のビット{0,1}を双曲空間 $\mathbb{H}^d$ に置き換えた



(a) dog and car



埋め込みの幾何と代数(研究紹介)

[Yoshikawa&Matsubara, ICLR, 2026]

双曲空間のデカルト積に ℓ_1 -直積計量を入れる

■ 概念の合成(属性の積集合)が可能

- 各空間でノルムが大きい方の埋め込みを採用($\equiv \max$)



(a) conjunctive prompt “a dog and a car”



(b) “max” of single-concept prompts
“a dog” and “a car”

データ多様体と測地線

多様体仮説を使ってデータ間の距離を定めたい

■ もっと明示的に距離を定めたい

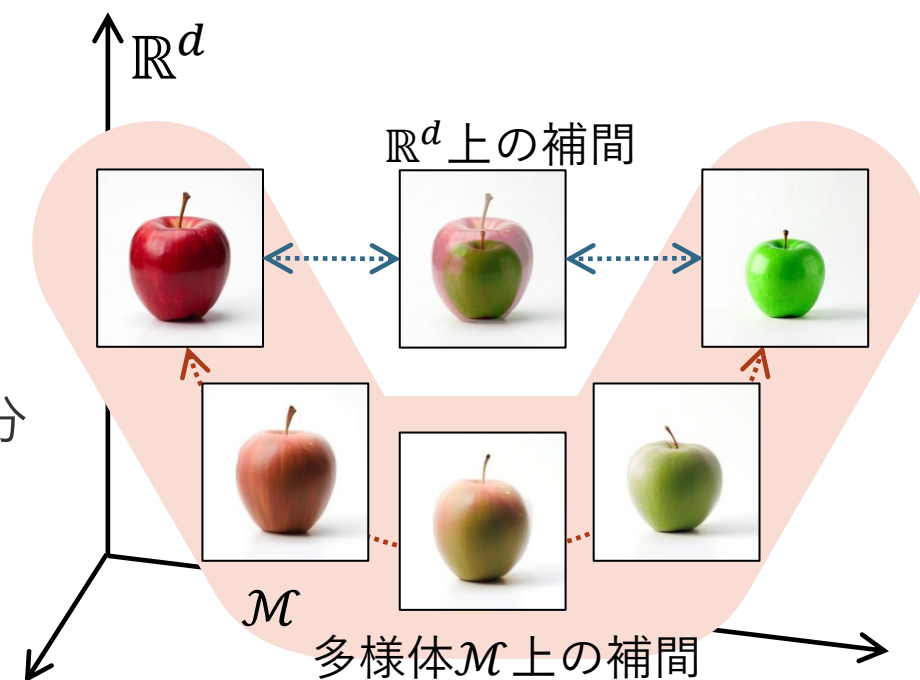
- ほとんど似た 2 枚なら誤差はわかりやすい
⇒ 局所的には $\mathbb{R}^d$ 上のユークリッド距離が良い
- 遠く離れた 2 枚はそうでもない
⇒ データ多様体 $\mathcal{M}$ 上の距離が欲しい

■ リーマン計量 g を導入する

- 2つの速度ベクトルの間に内積を定める
→ 速度ベクトルの長さを定める
 - ユークリッド空間だと普通の内積
- 適当な経路 $\gamma(t) \subset \mathcal{M}$ の長さは、速度ベクトルの長さの積分

$$\ell[\gamma] = \int_0^1 \sqrt{g(v(t), v(t))} dt \text{ for } v(t) = \gamma'(t)$$

- 2 点間の距離は最短経路(測地線)の長さ



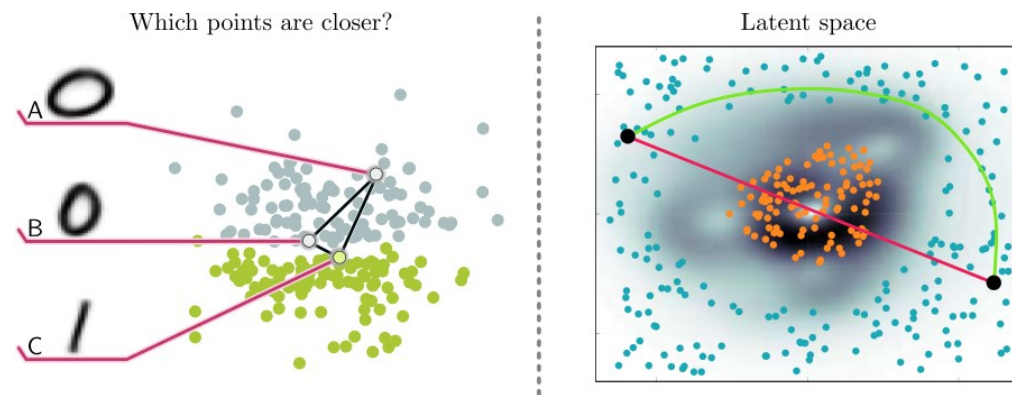
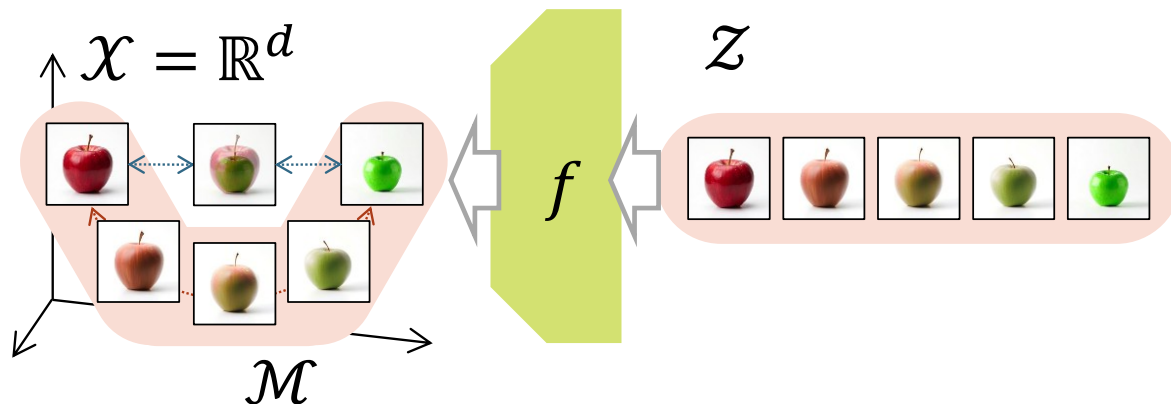
[松原, 数理科学, 2025]

データ多様体と測地線

潜在空間に距離を定める

■ 引き戻し計量

- 生成モデル $f: \mathcal{Z} \rightarrow \mathcal{M} (\subset \mathbb{R}^d)$
- データ空間 $\mathbb{R}^d$ 上のユークリッド計量 g_E を考える
- 表現空間 $\mathcal{Z}$ 上の経路 γ をデータ多様体 $\mathcal{M}$ 上の経路 $f_*\gamma = f(\gamma)$ に押し出す (pushforward)
 - 速度ベクトル $\gamma'(t)$ も押し出されて $f_*(\gamma'(t)) = f(\gamma)'$ となる
 - 押し出された経路 $f_*\gamma$ の長さ, 速度ベクトル $f(\gamma)'$ の長さを g_E で定める
- 長さが整合するような表現空間 $\mathcal{Z}$ 上のリーマン計量 f^*g_E が定まる (引き戻し (pullback))
- 表現空間 $\mathcal{Z}$ における測地線 = データ多様体 $\mathcal{M}$ 上の最短経路



[Arvanitidis et al., ICLR, 2018]

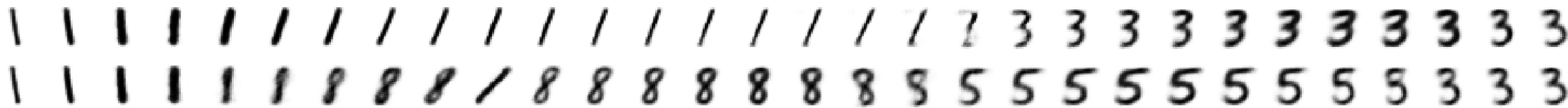
データ多様体と測地線

潜在空間に距離を定める

■ 引き戻し計量

- 生成モデル $f: \mathcal{Z} \rightarrow \mathcal{M} (\subset \mathbb{R}^d)$
- データ空間 $\mathbb{R}^d$ 上のユークリッド計量 g_E を考える
- 表現空間 $\mathcal{Z}$ 上の経路 γ をデータ多様体 $\mathcal{M}$ 上の経路 $f_*\gamma = f(\gamma)$ に押し出す (pushforward)
 - 速度ベクトル $\gamma'(t)$ も押し出されて $f_*(\gamma'(t)) = f(\gamma)'$ となる
 - 押し出された経路 $f_*\gamma$ の長さ, 速度ベクトル $f(\gamma)'$ の長さを g_E で定める
- 長さが整合するような表現空間 $\mathcal{Z}$ 上のリーマン計量 f^*g_E が定まる (引き戻し (pullback))
- 表現空間 $\mathcal{Z}$ における測地線 = データ多様体 $\mathcal{M}$ 上の最短経路

直線的な移動



測地線に沿った移動

[Chen et al., AISTATS, 2018]

データ多様体と測地線

低次元空間を使わずデータ間の距離を定めたい

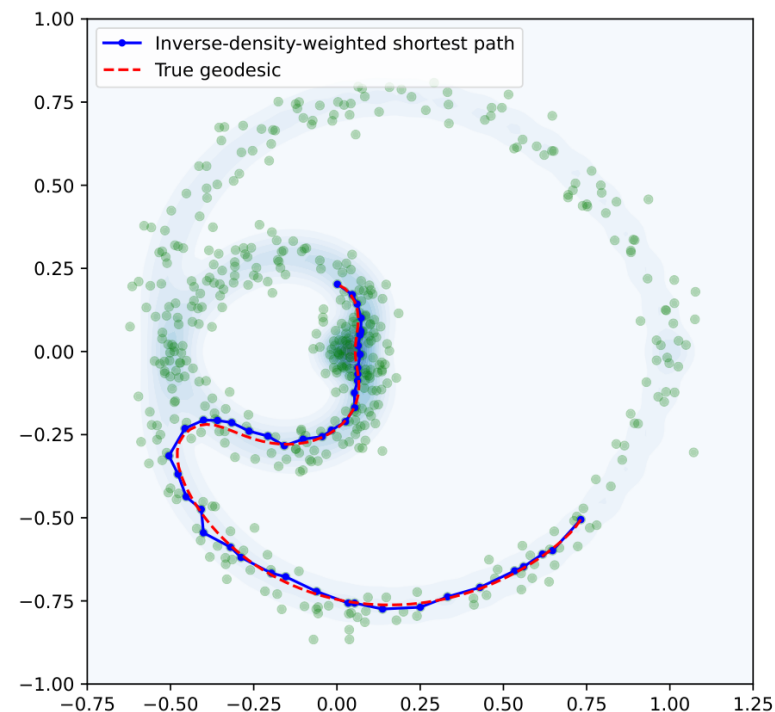
■ 低次元表現空間がない生成モデルが人気

- 表現空間の次元数が同じ：拡散モデル，フローベースモデル
- 表現空間がない：エネルギーベースモデル
- 潜在変数をたくさん使う：最新のGAN
- 実データはノイズによって完全な低次元ではない

■ データ空間の計量はユークリッドでないかも

- データ多様体上は短く，多様体から離れた場所は長い
 - 計量の例： $g = g_E/p(x)$
 - 密度が高いところ＝データ多様体と考える
- しかし，経験的に，密度高いところ＝低品質
 - リアルなノイズや雑味のない，のぺったした画像に
 - 多様体に近づきすぎてはいけない

[Karczewski et al., ICML, 2025]



[Sorensen et al., ICML, 2025]

[Yu et al., CVPR, 2025]

データ多様体と測地線(研究紹介)

[Saito&Matsubara, ECCV, 2026]

低次元空間を使わずデータ間の距離を定めたい

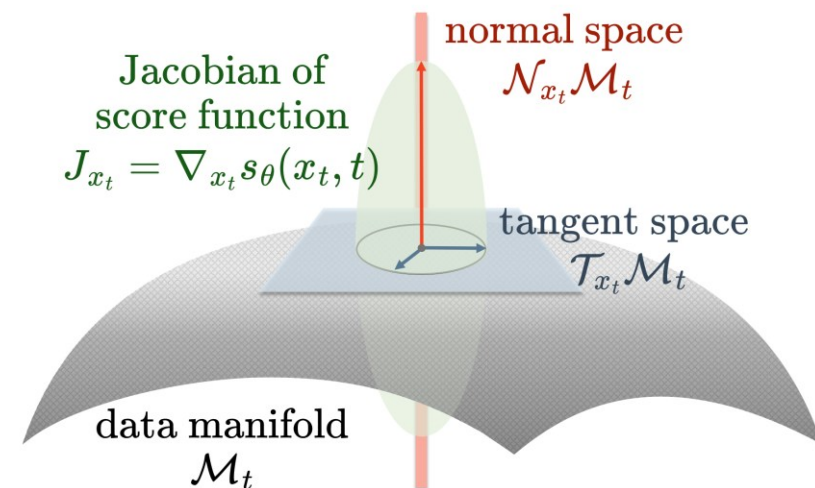
■ 多様体から適切な距離を保つ計量

- 対数確率密度 $\log p$ のヘッセ行列 H_x の固有ベクトルは多様体法空間方向に伸び、接空間方向に潰れる

- ヘッセ行列で計量を定義 $g = H_x^\top H_x$

⇒ 確率密度がなるべく変化しない経路を通る

- 学習するのはスコア $s(x) = \nabla_x \log p(x)$ なので $H_x = J_x = \frac{\partial s}{\partial x}$



データ多様体と測地線(研究紹介)

[Saito&Matsubara, ECCV, 2026]

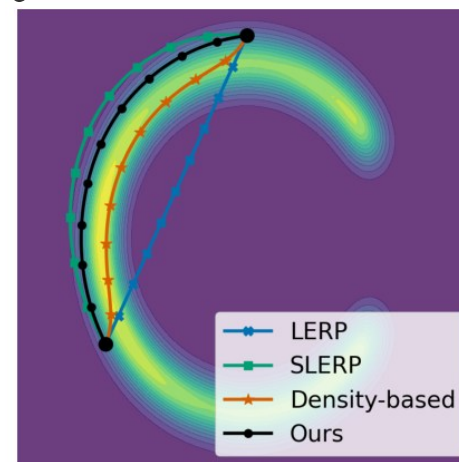
低次元空間を使わずデータ間の距離を定めたい

■ 多様体から適切な距離を保つ計量

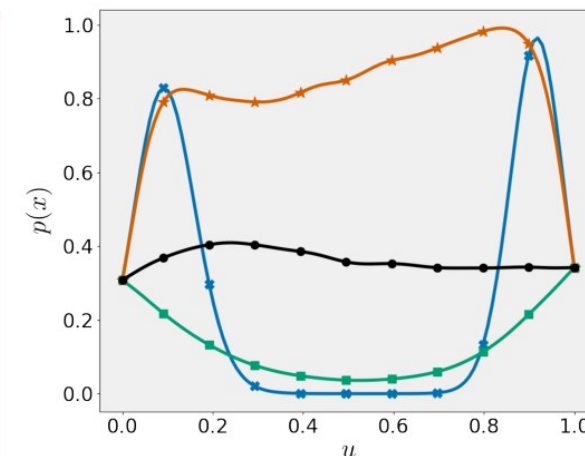
- ヤコビ行列で計量を定義 $g = J_x^T J_x$
 - 対数確率密度の勾配(スコア) s の空間におけるユークリッド計量の引き戻し $g = s^* g_E$

⇒ スコア s という特徴量なるべく保つ経路を通る

- 滑らかな変化が実現



(a)



(b)

LERP



GeoDiff



Ours



$x_0^{(0)}$

$\hat{x}_0^{(0)}$

$\hat{x}_0^{(u)}$

$\hat{x}_0^{(1)}$

$x_0^{(1)}$

データ多様体と測地線(研究紹介)

[Saito&Matsubara, ECCV, 2026]

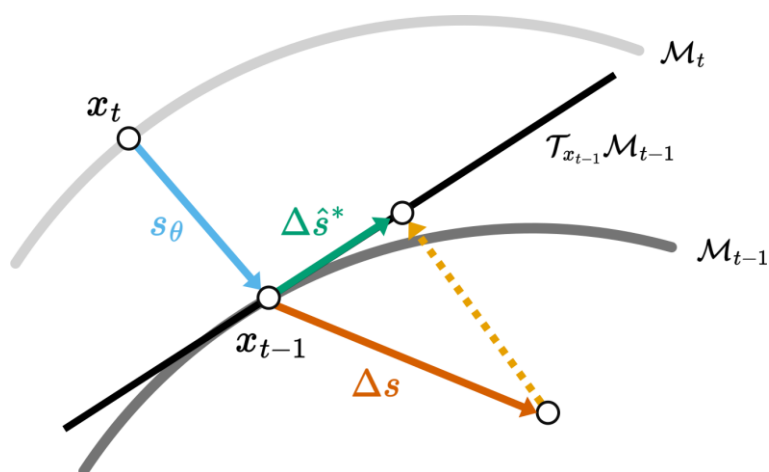
リーマン計量を使ってデータ多様体を定める

■ 忠実性と品質のトレードオフ

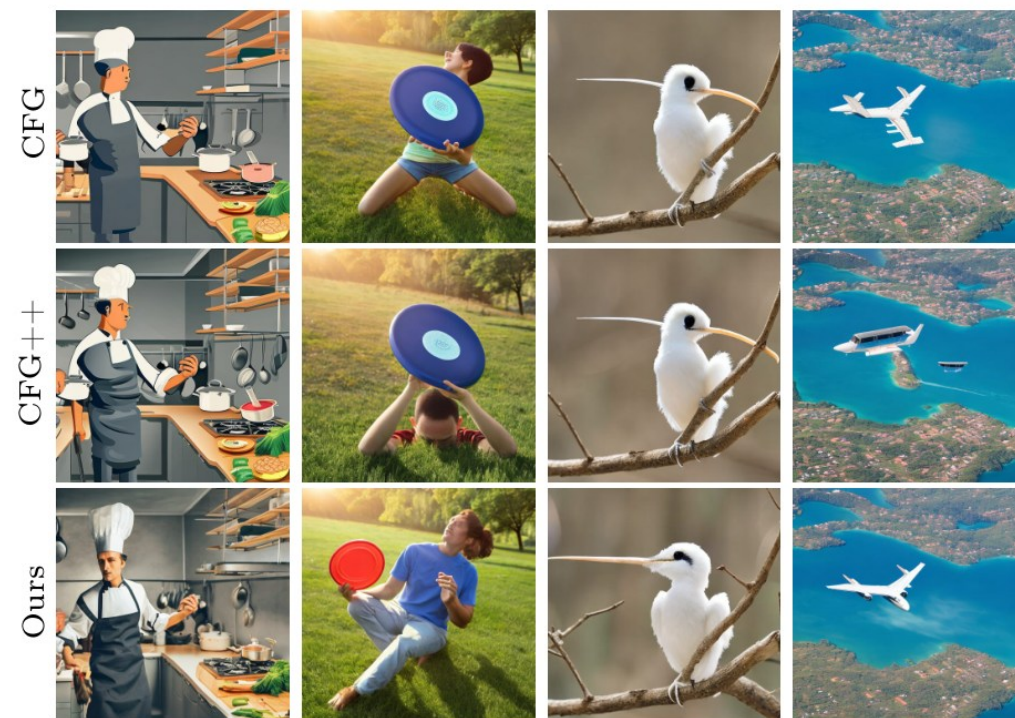
- CFGは更新幅を増やしてテキスト忠実性を向上
- 多様体から外れるの品質が落ちがち

■ 更新幅を接空間上に射影する

- 忠実度そのまま品質向上



	$w = 5.0$		$w = 7.5$		$w = 12.5$	
	FID ↓	CLIP ↑	FID ↓	CLIP ↑	FID ↓	CLIP ↑
CFG	<u>11.69</u>	0.313	14.29	0.314	<u>17.28</u>	0.315
CFG++	11.87	0.313	<u>13.98</u>	0.314	17.76	0.315
Ours	11.53	0.313	13.81	0.314	16.04	0.315



“A man is cooking in a crowded kitchen”

“A person is in the grass holding a frisbee”

“A small white bird with a long beak on a branch”

“A plane flies over water with two islands nearby”

まとめ

データの幾何学的構造と深層学習

- 幾何学的深層学習
 - ネットワークの設計で構造を強制しよう
- 幾何学的な表現学習^{1,2}
 - 損失関数に構造を入れて学習しよう
- データ多様体
 - 学習済みモデルから構造を取り出そう