大規模言語モデルにおいて数値属性間で共有されるスケーリングベクトルの解析とその応用

峰岸剛基 [1,*] 高木洋羽 [1,*] 木澤翔太 [1,*] 助田一晟 [1] 谷中 瞳 [1]
[1] 東京大学
minegishi@weblab.t.u-tokyo.ac.jp,
{hirohane, shota-kizawa, sukeda-issei006, hyanaka}@g.ecc.u-tokyo.ac.jp

概要

大規模言語モデル（LLM）の利活用が進む一方で，その推論過程や内部表現は不透明である．本研究では，LLM の内部表現における数値属性の構造に着目し，部分的最小二乗法（PLS）を用いた probing を通じて，異なる属性間で共通する「数値を変調する方向成分（スケーリングベクトル）」の存在を明らかにする．さらに，異なる属性間の交絡は，特定属性への介入操作や few-shot プロンプティングによる出力に副作用を及ぼすことを実験的に示す．この結果は，LLM の内部表現に対する解釈可能性と実用上の制御手法を結び付け，LLM の出力の公平性や頑健性の研究に貢献しうる示唆を提供する．

1 はじめに

近年，大規模言語モデル（LLM）は自然言語処理や文章生成において飛躍的な性能向上を遂げ，その応用範囲は急速に拡大している．しかし，LLM の推論過程や内部表現は依然としてブラックボックスであり，その信頼性や説明可能性に課題が残る．そのため，モデル内部表現の構造や振る舞いを解明する**機械論的解釈可能性**が注目されている [1, 2, 3].

probing では，線形分類器等を用いてモデルの内部表現を解析し，その表現が特定の概念をどの程度エンコードしているかを評価する [4]．この手法により，LLM が品詞の区別 [4] や，特定の名詞か否か [5, 6]，さらには時間や空間などの数値的属性 [7, 8] といった内部表現を獲得していることが報告されている．また，活性化空間内で特定概念を表現する概念ベクトルを取り出し，それを用いて LLM の推論時にモデルの内部に介入して出力を変調できる [9].

一方で，LLM の出力の制御手法としては，入力文字列（プロンプト）の設計による**プロンプトエンジニアリング** [10] が広く知られている．しかし，適切なプロンプトは状況に依存し，高度な技術を要する．また，知識の矛盾による混乱 [11] や悪意あるプロンプトによる攻撃 [12, 13] など，LLM のプロンプトに対する頑健性には問題がある．

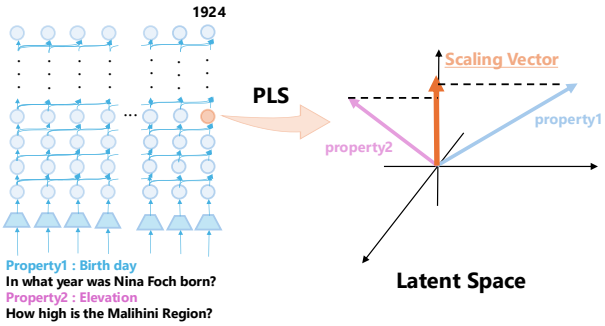


図 1: 本研究の分析手法の概要

プロンプトエンジニアリングによる LLM 制御に課題が残るが，他方でモデルの内部に介入する手法においても，回答の有益性が下がるリスク [9] や，特定の概念への介入操作が他の概念にも影響を与える副作用 [8] といった悪影響が観察されている．特に副作用は，LLM が異なる概念間の意味的な関係性を内部表現において捉えていることとの関連が指摘されている [6] が，概念間の交絡が LLM の安定した制御を困難にする懸念がある．

本研究では，LLM における概念の交絡による副作用を解析するため，異なる**数値属性**間における表現の転移可能性を体系的に調査する．具体的には，probing により LLM の活性化空間において異なる属性を跨いだ「数値を変調する方向成分（**スケーリングベクトル**）」が存在することを実験的に示し，そこで得られた概念ベクトルを用いて他の属性への介入が可能であることを明らかにする (図 1)．加えて，プロンプトエンジニアリング技術の一つである

* 第一著者から第三著者までの貢献は等しい．

few-shot プロンプティングにおいて，プロンプト内に複数の概念が含まれると概念の交絡による副作用が生じるかを確認する．これらの実験結果は，LLM の内部表現における概念間の構造を解釈するだけでなく，LLM の制御においても概念間の交絡を考慮すべき可能性を取り上げ，プロンプトエンジニアリングに対する実用的な知見を提供する．

2 関連研究

2.1 LLM への probing 手法

言語モデルに対する probing の初期の研究では，ロジスティック回帰による二値分類器を用いて言語的概念が内部表現に埋め込まれているか検証された [4]. 近年は，特定の名詞概念の分類 [5, 6] や，時空間概念に関する回帰 [7, 8] によって，LLM が実世界を反映した内部表現を獲得していることが示唆されている．これらの分析はいずれも特定概念に対する probing であり，本研究では多概念間の関係も扱う．

2.2 内部表現への介入とその副作用

probing から概念ベクトルを得てモデルの内部表現を操作する推論時介入（inference-time intervention）[9] に代表される**介入**手法も近年注目されている．しかし，特定概念への介入操作は，他概念にも影響を与える副作用を伴う [8]．介入を LLM の制御に用いる際にこの副作用は予期せぬ出力をもたらす懸念があるが，その分析は十分には行われていない．

2.3 プロンプトの脆弱性

LLM の制御手法として，プロンプトエンジニアリング [10] は広く活用されている．例えば，プロンプト内に質問と回答の例をいくつか含める few-shot プロンプティング [14] は推論能力の向上や出力形式の整形に有効である．しかし，モデルの内部知識とプロンプトによる外部知識の矛盾による混乱 [11] や，悪意ある文字列をプロンプトに含めることで LLM から望ましくない出力を引き出すプロンプトインジェクション [15] などの攻撃手法から観察されるように，プロンプトに依存した制御には脆弱性が存在する．本研究では，介入で観察された副作用に着想を得て，few-shot プロンプティングとしてプロンプト内に含めた例が，それ自体には悪意がないにも関わらず，副作用を及ぼして LLM の出力に干渉するかを検証する．

3 手法

数値属性の解析には，質問文とそれに対する数値の回答の組からなるデータセットを利用する．本研究では，まず probing により LLM の概念間の内部表現の性質を調べる．その後，LLM 内の概念間の交絡による副作用を，介入と few-shot プロンプティングにより確認し，それらが一貫するかを検証する．

3.1 PLS を用いた数値属性の分析方法

LLM の潜在次元 h, 質問文を LLM に入力した際にある層が出力する特定トークンに対応する内部表現 $x \in \mathbb{R}^h$ とする．サンプル数 n のデータセットを用いた probing では，データ行列 $X = [x_1, \cdots x_N]^\mathsf{T} \in \mathbb{R}^{n \times h}$ を目的変数 $Y \in \mathbb{R}^n$ に回帰する．多くの場合で h は大きい ($> n$) ため，正則化 [7] や次元削減 [8] が必要である．数値属性に対する probing では，目的変数と共分散が最大になる活性化空間内の主成分を取り出す部分的最小二乗法（Partial Least Squares, PLS）[16] が有効である [8].

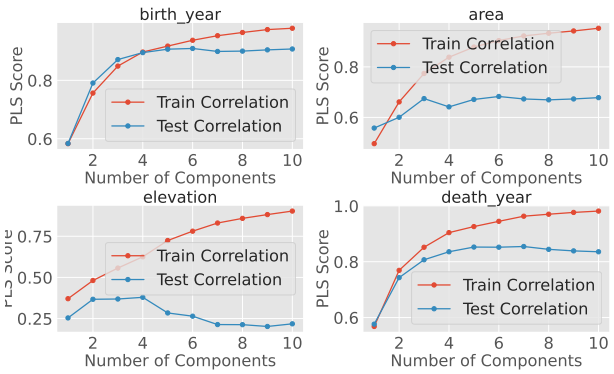
ランク k の PLS では，X 内で Y との共分散を最大化し予測に寄与する k 個の成分を取り出す変換行列 $W \in \mathbb{R}^{h \times k}$ を用いて，次元削減後の $Z = XW \in \mathbb{R}^{n \times k}$ を得る．同時に，X に関する係数 $P \in \mathbb{R}^{h \times k}$ s.t. $X \approx ZP^\mathsf{T}$ と Y に関する係数 $C \in \mathbb{R}^k$ s.t. $Y \approx ZC^\mathsf{T}$ も求める．この計算はライブラリ [17] によって効率的に行われる．当てはまりの良さは決定係数で評価する．

従来の probing の研究では原則一つの概念に関するデータセットで回帰を行うが，本研究では複数の数値属性の集合 $\mathcal{P}$ を考え，複数属性のデータセットを結合したものに対する PLS も扱う．

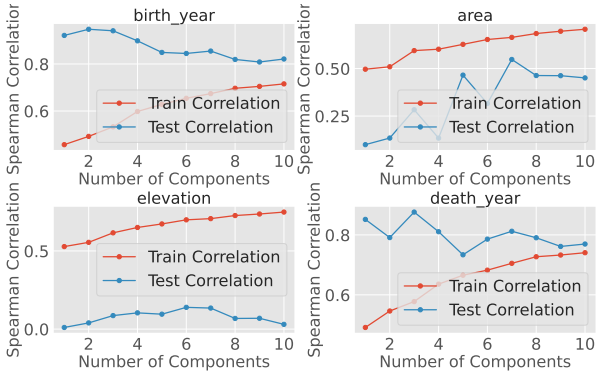
3.2 数値特性の介入方法

probing によって，LLM の活性化空間にて特定の概念を表す方向 $v \in \mathbb{R}^h$ が得られると，それを用いた LLM の推論時介入ができる [9, 8]．具体的には，ある層の特定トークンの内部表現 x に対して，介入重み $\alpha \in \mathbb{R}$ を用いて $x \leftarrow x + \alpha v$ の介入操作により，その概念を増幅・抑圧するような出力が得られる．

PLS による probing の場合には，活性化空間内で Y に関連する方向 $P_{:,j}$ ($j \in \{1, \cdots k\}$) を v として用いる．また，適切な α の設定が必要だが，ここでは αv が $\min(XW_{:j})P_{:j}$ と $\max(XW_{:j})P_{:j}$ を M 等分するように選び，α と LLM の出力値のスピアマン相関によりどれだけ介入されやすいかを評価する．



(a) 各属性で PLS を学習し，同じ属性を評価



(b) 各属性を**除いて**学習し，除かれた属性で評価

図 2: 質問文の内部表現で PLS を学習し，評価した際の決定係数.

3.3 Few-shot 例による概念の交絡の分析

ある属性 $p \in \mathcal{P}$ のデータセットの質問文をプロンプトに含めて LLM に入力すると，数値の回答が得られる．ここで，属性 $p' \in \mathcal{P}$ の情報を few-shot プロンプティングの例としてプロンプトに含めた際の副作用を調査する．p' に関する例示質問文と例示回答を few-shot プロンプティングの例としてプロンプトに加えて LLM に入力し，元の p に対する予測回答を得る．プロンプトの例を 付録の表 2 に示す．これを繰り返し，例示回答と予測回答のスピアマン相関で副作用を評価する．

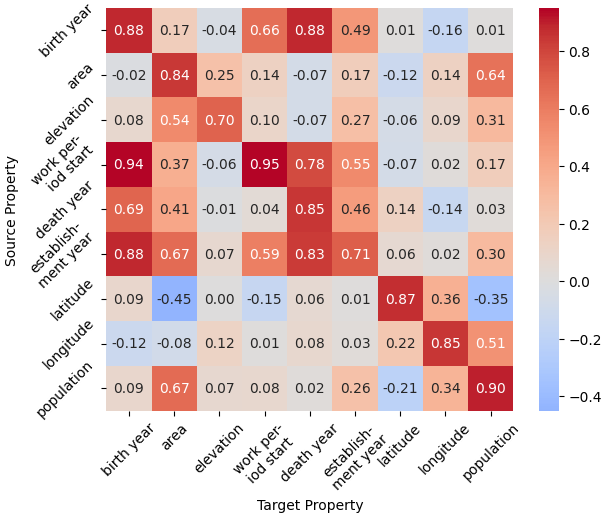


図 3: 各属性の内部表現で学習した PLS の予測と他の属性のもとの数量との相関.

4 実験

4.1 実験設定

Llama 3.1[18] の 8B モデルを解析対象とし，データセットは先行研究 [8] と同様に Wikidata[19] を使用する．数値属性の集合 $\mathcal{P}$ には 9 種類の属性（例：birth year，death year，elevation など）を用いる．詳細は付録の表 1に示す．各属性について訓練データ 1000 件，検証データ 100 件，テストデータ 1000 件からなる．時空間的概念に関する表現は入力層から中間層に至る途中で効率的に処理される [7, 8] ため，probing と介入には全 33 層のうち 10 層目を用いる．

probing には **Question** を用いた時の LLM の最後のトークンに対応する内部表現を解析の対象とする．また，目的変数 Y は各属性内にて Yeo-Johnson 変換を行い標準正規分布に近づけた．

介入時は，LLM に数値の出力を促すために

“{**Question**} `One word answer only:`”

を入力し，**Question** の末尾 2 トークンとその次の 2 トークンに介入する．また計算コストの都合から，介入の評価にはランダムにサンプリングされたデータセットの一部を用いる．介入の重みの刻み $M = 21$ とし，介入方向 v の添え字 j は，検証データから 30 件を用いてスピアマン相関の絶対値が最も大きい方向を採用する．それを用いて，テストデータ 100 件で最終的な相関を評価する．

4.2 PLS による数値属性の probing

まず，属性 p の学習データに PLS をフィッティングし，その属性 p のテストデータで評価する．性能は決定係数で評価する．PLS のランク k を変えながら PLS を行った結果を図 2(a) に示す．多くの属性で PLS の性能は良好である．この結果は Llama 2 に対する先行研究 [8] の結果と一貫している．

次に，属性 p を除外した $\mathcal{P} \setminus \{p\}$ で PLS を学習し，属性 p のテストデータにより評価する．各属性内では Y は前処理済みだが，PLS が属性を跨いでも正

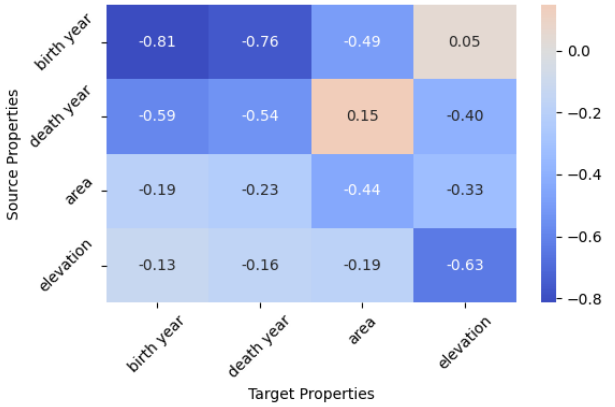


図 4: 各属性の内部表現の PLS から得られる概念方向から，目的属性への介入した際の相関.

図 5: few-shot の数値と LLM の回答の相関.

確な値を予測するとは限らない．そのため，評価指標は，全体的なずれの影響を差し引いて値の大小を比較するスピアマン相関を用いる．図 2(b) に，各 k に対する PLS の結果を示す．birth year や death year, area は，その属性のデータを PLS を学習に使っていないのにも関わらず値の大小を予測できる．この結果は，LLM の活性化空間内には複数の属性に共通して数字の大きさを表現するスケーリングベクトルが存在することを示唆する．

さらに，ある属性 p の学習データにより PLS をフィッティングし，他属性 $p' \in \mathcal{P} \setminus \{p\}$ のテストデータを用いてスピアマン相関を評価する．学習データに対してランク $k = 1, \cdots 10$ の PLS を行い，検証データに対して最良のランク k を選択した際のテストデータに対するスピアマン相関をプロットした結果を図 3に示す．特に birth year と death year は高い相関が得られ，目的属性の質問文の内部表現で学習しなくても他属性から値の大小を予測できる．また，付録図 6には全属性 $\mathcal{P}$ を結合したデータに対する PLS により LLM の活性化空間を次元削減して可視化した結果を示す．いくつかの属性は大小に関して共通した方向を持つことが確認できる．

4.3 介入とその副作用

図 4に一部属性における介入の効果を示す．非対角成分の副作用について，PLS による他属性への予測可能性 (図 3) が大きいものは介入の副作用も大きくなる一貫性が見られる．また，年に関する属性 birth year と death year は互いに介入されうる．特に，birth year が他属性に介入しやすいことと，他属性の PLS から birth year を予測しやすいことは，Llama3.1 が持つスケーリングベクトルと birth year の概念ベクトルが近い可能性を示唆する．

スケーリングベクトルは相関関係を捉えるだけでなく，因果的な操作ツールになり得る．特定の属性における介入が他の関連する数値属性に及ぼす副作用を考えると，スケーリングベクトルの考慮は LLM の解釈可能性だけでなく，制御可能性の向上に寄与する可能性がある．

4.4 Few-shot 例による知識の交絡

図 5 に，few-shot 例による知識の交絡の影響に関する実験結果を示す．図 3で強い相関のある属性 (birth year と death year) は few-shot で事例を与えた場合でも影響を受けやすい．この結果は，プロンプトインジェクションに関する先行研究で指摘されているようなプロンプトに対する鋭敏性が属性間に共通する方向成分（スケーリングベクトル）に起因している可能性を示唆し，より効果的かつ頑健なプロンプトを設計する上での実用的な指針を提供する．

5 おわりに

本研究では，LLM が活性化空間に持つ数値属性の構造を題材に，異なる属性間に共通する「スケーリングベクトル」を分析した．これにより，LLM が数値をどう扱い，概念間の関係性をどのように捉えているのかを明らかにすることを試みた．とくに，属性間の交絡がプロンプト設計や出力制御に与える影響を実験的に示したことは，より賢い LLM の使い方を模索するうえでの足がかりになる．今後の方向性として，より副作用が問題とされる文脈．例えば LLM の持つ交差バイアス [20] のような公平性の検証やプロンプトを介した攻撃における内部状態の解析への応用が挙げられる．

謝辞

本研究は JSPS 科研費学術変革領域研究（B）「ナラティブ意識学」JP24H00809 の支援を受けたものである.

参考文献

[1] Leonard Bereska and Efstratios Gavves. Mechanistic Interpretability for AI Safety–A Review. **arXiv preprint arXiv:2404.14082**, 2024.

[2] Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability, 2023.

[3] Chandan Singh, Jeevana Priya Inala, Michel Galley, Rich Caruana, and Jianfeng Gao. Rethinking interpretability in the era of large language models, 2024.

[4] Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. **Computational Linguistics**, Vol. 48, No. 1, pp. 207–219, 04 2022.

[5] Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In **The Eleventh International Conference on Learning Representations**, 2023.

[6] Haiyan Zhao, Heng Zhao, Bo Shen, Ali Payani, Fan Yang, and Mengnan Du. Beyond Single Concept Vector: Modeling Concept Subspace in LLMs with Gaussian Distribution. **arXiv preprint arXiv:2410.00153**, 2024.

[7] Wes Gurnee and Max Tegmark. Language models represent space and time. In **The Twelfth International Conference on Learning Representations**, 2024.

[8] Benjamin Heinzerling and Kentaro Inui. Monotonic representation of numeric attributes in language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, **Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)**, pp. 175–195, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[9] Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, **Advances in Neural Information Processing Systems**, Vol. 36, pp. 41451–41530. Curran Associates, Inc., 2023.

[10] Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. A systematic survey of prompt engineering in large language models: Techniques and applications. **arXiv preprint arXiv:2402.07927**, 2024.

[11] Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. Adaptive Chameleon or Stubborn Sloth: Revealing the Behavior of Large Language Models in Knowledge Conflicts. In **The Twelfth International Conference on Learning Representations**, 2024.

[12] Yi Zeng, Hongpeng Lin, Jingwen Zhang, Diyi Yang, Ruoxi Jia, and Weiyan Shi. How Johnny Can Persuade LLMs to Jailbreak Them: Rethinking Persuasion to Challenge AI Safety by Humanizing LLMs. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, **Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 14322–14350, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[13] Cem Anil, Esin Durmus, Nina Rimsky, Mrinank Sharma, Joe Benton, Sandipan Kundu, Joshua Batson, Meg Tong, Jesse Mu, Daniel J Ford, Francesco Mosconi, Rajashree Agrawal, Rylan Schaeffer, Naomi Bashkansky, Samuel Svenningsen, Mike Lambert, Ansh Radhakrishnan, Carson Denison, Evan J Hubinger, Yuntao Bai, Trenton Bricken, Timothy Maxwell, Nicholas Schiefer, James Sully, Alex Tamkin, Tamera Lanham, Karina Nguyen, Tomasz Korbak, Jared Kaplan, Deep Ganguli, Samuel R. Bowman, Ethan Perez, Roger Baker Grosse, and David Duvenaud. Many-shot jailbreaking. In **The Thirty-eighth Annual Conference on Neural Information Processing Systems**, 2024.

[14] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, **Advances in Neural Information Processing Systems**, Vol. 33, pp. 1877–1901. Curran Associates, Inc., 2020.

[15] Yupei Liu, Yuqi Jia, Runpeng Geng, Jinyuan Jia, and Neil Zhenqiang Gong. Formalizing and benchmarking prompt injection attacks and defenses. In **33rd USENIX Security Symposium (USENIX Security 24)**, pp. 1831–1847, Philadelphia, PA, August 2024. USENIX Association.

[16] Svante Wold, Michael Sjöström, and Lennart Eriksson. PLS-regression: a basic tool of chemometrics. **Chemometrics and Intelligent Laboratory Systems**, Vol. 58, No. 2, pp. 109–130, 2001. PLS Methods.

[17] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, Vol. 12, pp. 2825–2830, 2011.

[18] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The Llama 3 herd of models. **arXiv preprint arXiv:2407.21783**, 2024.

[19] Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. **Commun. ACM**, Vol. 57, No. 10, p. 78–85, September 2014.

[20] John Lalor, Yi Yang, Kendall Smith, Nicole Forsgren, and Ahmed Abbasi. Benchmarking intersectional biases in NLP. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, editors, **Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 3598–3609, Seattle, United States, July 2022. Association for Computational Linguistics.

A　多属性データに対する PLS で次元削減した活性化空間

図 6に，(birth year, death year, establishment year) と (latitude, longtitude) の多属性データに対する PLS で得た活性化空間のプロットを示す. 各点の色は属性の違いを大きさは数値の大きさを表す. 図 3から PLS で互いに予測できている birth year と death year などは数値の大小に対して共通する方向成分を持つ．また図 3の相関の小さい latitude や longtitude などはそのような方向を持たない.

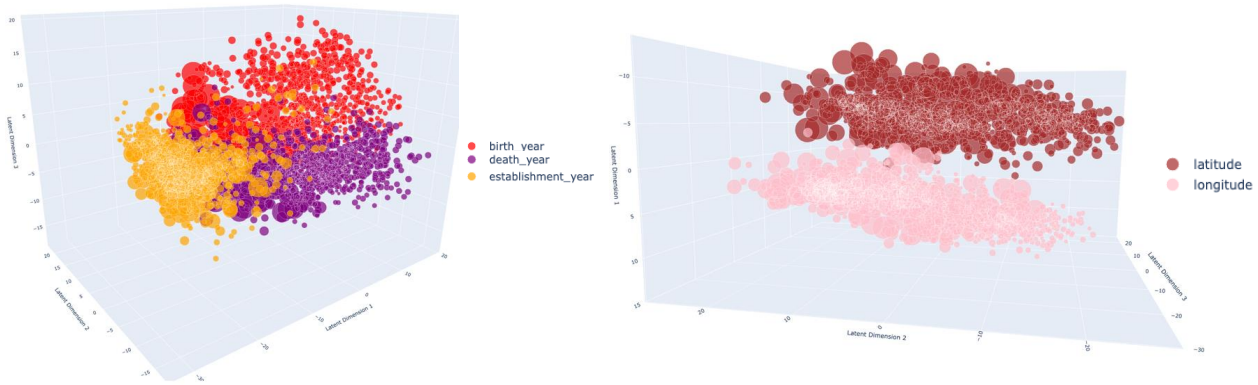


図 6: PLS による次元削減後の活性化空間

B　データセットの詳細

表 1: 使用するデータセットのサンプル．属性は birth year, death year, elevation, area, work period start, establish year, latitude, longtitude の 9 つからなる. ここでは代表的な 3 つを載せる.

Property	Question	Answer
birth year	In what year was Nina Foch born?	1924
death year	In what year did Carlos Santiago Mérida die?	1984
elevation	How high is Malihini Region?	191

C　Few-shot 実験のプロンプトの詳細

表 2: death year に関する指示に対して， elevation に関する few-shot 事例を入れる場合のプロンプトのサンプル. 実験では指示文は固定し，few-shot 事例を変えたときにどの程度 LLM の回答が変動するかを測る.

Q. How high is Erie? A. 223 Q. In what year did Otto Ohlendorf die? A.
Q. How high is Aydın? A. 65 Q. In what year did Otto Ohlendorf die? A.

D　介入重みを変更した時の出力の変化

表 3: LLM に対して "In what year was Diana Mosley born? One word answer only:" を入力し，birth year に対する PLS で獲得した介入方向 v と Weight パラメータ α を用い，α を変更して介入した際のそれぞれの出力結果.

α	-21.34	-19.20	-17.07	-14.94	-12.80	-10.67	-8.53	-6.40	-4.27	-2.13	
出力	1960	1960	1960	1953	1953	1938	1938	1933	1920	1920	
	0.00	1.87	3.75	5.62	7.49	9.37	11.24	13.11	14.98	16.86	18.73
	1910	1910	1882	1870	1850	1850	1850	1850	1848	1848	1848