Identifiable attribution maps using regularized contrastive learning

Steffen Schneider
EPFL
Geneva, CH

Rodrigo González Laiz
EPFL
Geneva, CH

Markus Frey
EPFL
Geneva, CH

Mackenzie W. Mathis[*]
EPFL
Geneva, CH

Abstract

Gradient-based attribution methods aim to explain decisions of deep learning models, but so far lack identifiability guarantees. Here, we propose a method to generate attribution maps with identifiability guarantees by developing a regularized contrastive learning algorithm trained on time series data with continuous target labels. We show theoretically that our formulation of hybrid contrastive learning has favorable properties for identifying the Jacobian matrix of the data generating process, and is unable to overfit to random training distributions. Empirically, we demonstrate robust approximation of the ground-truth attribution map on synthetic data, and significant improvements across previous attribution methods based on feature ablation, Shapley values, and other gradient-based methods. Our work constitutes a first example of identifiable inference of attribution maps, and opens avenues for improving future attribution tools and better understanding neural dynamics and neural networks.

1 Introduction

Distilling knowledge from data is a core tenet of science. After pre-processing raw data, we want to abstract relationships in the experimentally observed data to observed variables. In the case of neuroscience this could be the raw neural signal and the behavior of the animal [34, 10]. Often times linear methods (such as GLMs [15]) are used for interpretability, even though the underlying data did not necessarily arise from linear processes. Yet, non-linear methods are difficult to interpret [5, 25].

In machine learning, especially in computer vision, many algorithms exist for explaining the decisions of trained (non-linear) neural networks, often on classification tasks [25, 1, 29, 33, 17, 31, 14]. In particular, gradient-based attribution methods have shown empirical success, but can be computationally costly and/or lack theoretical grounding [31, 14], which ultimately limits their utility and scope in scientific applications that benefit from theoretical guarantees.

Contrastive learning recently showed promise in its performance for learning representations while providing theoretical guarantees about its representations [8, 9, 26, 37]. In this work, we aim to unify the empirical performance of gradient-based attribution methods for generating explanations of large scale datasets with complex non-linear relationships between their variables. We propose a contrastive learning method that provably identifies an attribution map underlying the data. Our framework is depicted in Fig. 1, and contributes the following: (1) We formulate an estimation algorithm for global attribution maps based on contrastive learning; (2) We show identifiability guarantees for the (global) attribution map and verify our theory on synthetic datasets.

[*] Correspondence: mackenzie.mathis@epfl.ch and steffen.schneider@epfl.ch

NeurIPS 2023 Workshop: Self-Supervised Learning – Theory and Practice

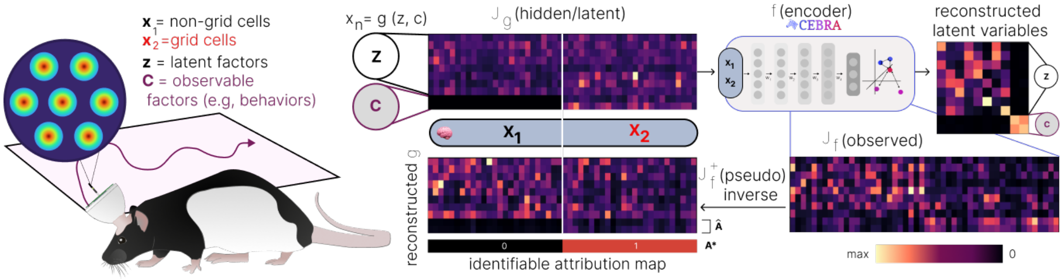


Figure 1: **Identifiable attribution maps for time-series data.** Using time-series data (such as neural data recorded during navigation, as depicted), our inference framework estimates the ground-truth Jacobian matrix $\mathbf{J_g}$ (i.e., $\mathbf{g}$ is the observed neural data linked to latents $\mathbf{z}$ and $\mathbf{c}$, where $\mathbf{c}$ is the explicit behavioral latent that would be linked to grid cells) by identifying the inverse data generation process up to a linear indeterminacy $\mathbf{L}$. Then, we estimate the Jacobian $\mathbf{J_f}$ of the encoder by minimizing a generalized InfoNCE objective. Inverting this Jacobian $\mathbf{J_f^+}$, which approximates $\mathbf{J_g}$, allows us to construct the attribution map.

2 Identifiability of Attribution Maps with Regularized Contrastive Learning

Throughout the paper, we will use a notion of attribution maps grounded in the causal structure of the data generating process. We assume that observations $\mathbf{x} \in \mathcal{X}$ are generated by an injective generative process (mixing function) $\mathbf{g} : \mathcal{Z} \to \mathcal{X}$, where $\mathcal{X} \subseteq \mathbb{R}^D$ is the space of observations and $\mathcal{Z} \subseteq \mathbb{R}^d$ is the space of latent factors and d denotes the number of factors. We have $d < D$.

Definition 1 (Data generating process)**.** We assume a non-linear ICA problem with a mixing function $\mathbf{g} : \mathcal{Z} \to \mathcal{X}$ mapping *parts* of the input factors $\mathbf{z} := [\mathbf{z}_1; \dots; \mathbf{z}_G] \in \mathbb{R}^d$ onto outputs, $x_i = g_i(\mathbf{z}) = g_i([\mathbf{z}_j]_{j \in P_i})$. P_i is an index set, and $j \in P_i$ implies that factor $\mathbf{z}_j \in \mathbb{R}^{d_i}$ is used to generate the output x_i. We further assume that maximally one of the parts is not observable. All other parts are observable through bijective maps γ_i s.t. $\mathbf{z}_i = \gamma_i(\mathbf{c}_i)$, where $\mathbf{c}_i$ then denotes an observable factor.

For application of attribution methods to $\mathbf{g}$, we need additional structure in the data generating process. Specifically, we are interested in how the factors $\mathbf{z}$ are *connected* to the output variables $\mathbf{x}$ by means of any non-linear mapping. This gives rise to the following definition of the attribution map:

Definition 2 (Ground-truth attribution maps)**.** Let $\mathbf{g}$ be the mixing function. For all $\mathbf{x} := \mathbf{g}(\mathbf{z})$ in the support of $p(\mathbf{z})$, the ground-truth attribution map $\mathbf{A}[\mathbf{g}]$ has values

$$A[\mathbf{g}]_{ij} = \begin{cases} 1 & \text{if} \quad \dfrac{\partial g_i(\mathbf{z})}{\partial z_j} \neq 0 \quad \exists \mathbf{z} \in \mathcal{Z} \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

and is specified through the index sets $P_1, \dots, P_G$ defined in Def. 1.

Under these definitions, we aim to identify the attribution map using a suitable representation learning algorithm. We require two components: First, the algorithm needs to be able to identify the latent and observable factors of our data generation process and it is well-studied that contrastive learning algorithms have this property [9, 26, 37]. In the following, let us call p the positive and q the negative sample distribution. We call $(\mathbf{x}, \mathbf{x}^+)$ a positive pair, and all $(\mathbf{x}, \mathbf{x}_i^-)$ negative pairs. The function $\mathbf{f} := [\mathbf{f}_1; \dots; \mathbf{f}_G]$ is the feature encoder that maps samples into an embedding space, and we apply similarity metrics ϕ_i to the different parts of this feature encoder, abbreviated as $\psi(\mathbf{x}, \mathbf{y}) := \phi(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{y}))$. The minimizer of the generalized InfoNCE [35, 26] objective

$$\mathcal{L}_N[\psi] = \underset{\substack{\mathbf{x} \sim p(\mathbf{x}),\, \mathbf{x}^+ \sim p_i(\mathbf{x}^+|\mathbf{x}), \\ \mathbf{x}_1^- \dots \mathbf{x}_N^- \sim q(\mathbf{x}^-|\mathbf{x})}}{\mathbb{E}} \left[-\psi(\mathbf{x}, \mathbf{x}_i^+) + \log \sum_{j=1}^{N} e^{\psi(\mathbf{x}, \mathbf{x}_j^-)} \right], \tag{2}$$

is $\psi(\mathbf{x}, \mathbf{y}) = \log p(\mathbf{y}|\mathbf{x})/q(\mathbf{y}|\mathbf{x}) + C(\mathbf{x})$ and identifies the ground truth latents up to a linear transform for suitable choice of ϕ, p and q, $\mathbf{f}(\mathbf{g}(\mathbf{z})) = \mathbf{L}\mathbf{z}$ [26]. Importantly, we can separate a meaningful fit from random data as expressed in:

Theorem 1. *Assume ψ^* is a minimizer of the generalized InfoNCE loss under the ICA problem in Def. 1 for $G = 1$ parts in the limit $N \to \infty$. Assume the observable factors $\mathbf{c}$ with $\mathbf{z} = \gamma(\mathbf{c})$ are independent of $\mathbf{z}$. Then, $\psi^* = const.$ is the trivial solution with $\lim_{N \to \infty} \mathcal{L}_N[\psi^*] = \log N$.*

Proof Sketch. The minimizer of the contrastive learning objective is $\psi(\mathbf{x},\mathbf{y}) = \log p(\mathbf{y}|\mathbf{x})/q(\mathbf{y}|\mathbf{x}) + C(\mathbf{x})$. Assume that the latents $\mathbf{z}$ are independent from the used labels $\mathbf{c}$, then we have $\psi(\mathbf{x},\mathbf{y}) = C(\mathbf{x}) = \psi^*(\mathbf{x})$ independent of $\mathbf{x}$. Inserting into the objective functions gives $\mathcal{L}[\psi^*] = \log N$. The full proof is given in Appendix A.3. □

To identify the ground-truth attribution map, we apply this learning scheme to each partition of the latent variables. In addition, we need to regularize the Jacobian matrix [7] of the feature encoder to become minimal. With these constraints, we obtain the objective function for Regularized Contrastive Learning (RegCL) for all parts of the representation:

$$\mathcal{L}_N[\psi;\lambda] = \mathop{\mathbb{E}}_{\substack{\mathbf{x}\sim p(\mathbf{x}),\\ \mathbf{x}^+\sim p_i(\mathbf{x}^+|\mathbf{x})\ \forall i\in[G]\\ \mathbf{x}_1^-\dots\mathbf{x}_N^-\sim q(\mathbf{x}^-|\mathbf{x})}} \left[\sum_{i=1}^{G} \Big(-\psi_i(\mathbf{x},\mathbf{x}_i^+) + \log\sum_{j=1}^{N} e^{\psi_i(\mathbf{x},\mathbf{x}_j^-)}\Big) + \lambda\|\mathbf{J}_\mathbf{f}(\mathbf{x})\|_F^2\right]. \quad (3)$$

where $\mathbf{J}_\mathbf{f}(\mathbf{x})$ is the Jacobian of the feature encoder $\mathbf{f}$ optimized as part of ψ, $\|\cdot\|_F$ denotes the Frobenius norm and λ is a hyperparameter tuned based on the learning dynamics. λ is tuned to the highest value possible that still allows the InfoNCE component of the loss to stay at its minimum. Intuitively, this loss function solves G non-linear ICA problems using the single feature encoder $\mathbf{f}$ — for observable $\mathbf{z}_i = \gamma_i(\mathbf{c}_i)$ we leverage supervised contrastive learning with continuous labels [26], for the non-observable $\mathbf{z}_G$ we apply time-contrastive learning using the time-series structure [8, 26].

Theorem 2. *Consider a non-linear ICA problem with mixing function $\mathbf{g}$ mapping latent factors $\mathbf{z}$ to a signal space such that $\mathbf{x} = \mathbf{g}(\mathbf{z})$ according to Def. 1. Let $A_{ij} = 1\{\exists\mathbf{z} : |\partial g_i(\mathbf{z})/\partial z_j| \neq 0\}$ be the entries of the global attribution map $\mathbf{A}$ of the mixing function. Assume that in the limit $N \to \infty$, the differentiable feature encoder $\mathbf{f}$ minimizes the regularized contrastive loss (Eq. 3) on $p(\mathbf{z})$. Then, we identify the global attribution map through the pseudo-inverses $\mathbf{J}_\mathbf{f}^+(\mathbf{x})$,*

$$\mathbf{A} = \mathbf{J}_\mathbf{f}^+(\mathbf{x}) \odot \mathbf{L}(\mathbf{x}), \quad (4)$$

up to component-wise scaling $\mathbf{L}(\mathbf{x})$ of the entries.

Proof Sketch. The individual parts of the loss function result in $\psi(\mathbf{x},\mathbf{x}') = \log p_i(\mathbf{z}_i'|\mathbf{z}_i)/q(\mathbf{z}_i')$ from which a linear indeterminacy follows, $\mathbf{f}_i(\mathbf{g}(\mathbf{z})) = \mathbf{L}_i\mathbf{z}$. We can express the result as $\mathbf{f}(\mathbf{g}(\mathbf{z})) = \mathbf{L}\mathbf{z}$ where $\mathbf{L}$ is a block-diagonal matrix with zeros in its lower block triangular part. Hence, $\mathbf{L}^{-1}$ will have the same property. It then follows that $\mathbf{J}_\mathbf{f}(\mathbf{x})\mathbf{J}_\mathbf{g}(\mathbf{z}) = \mathbf{L}$ and since $\mathbf{J}_\mathbf{f}$ has minimum norm everywhere, $\mathbf{J}_\mathbf{f}^+(\mathbf{x})$ is the Moore-Penrose pseudo-inverse of $\mathbf{J}_\mathbf{g}(\mathbf{z})\mathbf{L}^{-1}$. Multiplication with $\mathbf{L}^{-1}$ does not alter the location of zero entries in $\mathbf{J}_\mathbf{g}(\mathbf{z})$, and hence thresholding $\mathbf{J}_\mathbf{f}^+(\mathbf{x})$ across samples $\mathbf{x}$ in the dataset is an estimator of the ground-truth attribution map. The full proof is given in Appendix A.4. □

3 Experimental verification

Experiment setup To verify our theory, we generate a synthetic dataset following Def. 1, cf. Appendix B.2 for details. We sample 10 different datasets with 100,000 samples, each with a different mixing function $\mathbf{g}$. The mixing functions consist of randomly initialized 3-layer MLPs [8] and we ensure injectivity by monitoring the condition number of each matrix layer, following previous work [8, 37]. Similar to Schneider et al. [26], the feature encoder $\mathbf{f}$ is an MLP with three layers followed by GELU activations [6], and one layer followed by a scaled $\tanh$ to decode the latents. We train on batches with 5,000 samples each. The first 2,500 training steps minimize the InfoNCE or supervised loss with $\lambda = 0$, we then ramp up λ to its maximum value over the following 2,500 steps, and continue to train until 20,000 total steps. We compute the R^2 for predicting the observable factors $\mathbf{c}$ from the feature space after a linear regression, and ensure that this metric is close to 100% for both our baseline and contrastive learning models to remove performance as a potential confounder.

As a comparison to previous work, we vary the training method (hybrid contrastive, supervised contrastive, standard supervised) and consider baseline methods for estimating the attribution maps (neuron gradients [31, 20], integrated gradients [30, 33], Shapely values [28, 14], and feature ablation [16]), which are commonly used algorithms in scientific applications [25, 16]. To compute these attribution maps, we leveraged the open source library captum [12]. We also compare regularized and non-regularized training. Hyperparameters are identical between training setups, the regularizer λ, and number of training steps are informed by the training dynamics.

attribution method	supervised none	 regularized	supervised contrastive none	 regularized	hybrid contrastive none	 regularized **(RegCL)**
Neuron Gradient	$79.2^{81.0}_{77.4}$	$93.0^{94.5}_{91.5}$	$80.6^{82.4}_{78.8}$	$86.7^{89.0}_{84.6}$	$79.2^{81.0}_{77.5}$	$88.0^{90.1}_{85.8}$
Feature Ablation	$83.1^{84.8}_{81.3}$	$88.5^{90.0}_{87.0}$	$84.0^{85.6}_{82.1}$	$84.7^{86.5}_{82.8}$	$82.9^{84.5}_{81.3}$	$85.2^{86.9}_{83.4}$
Integrated Gradients	$81.0^{82.7}_{79.2}$	$84.9^{86.6}_{83.1}$	$81.9^{83.7}_{80.2}$	$82.3^{84.3}_{80.5}$	$83.9^{85.6}_{82.1}$	$86.9^{88.8}_{84.9}$
Shapely, shuffled	$82.0^{83.7}_{80.3}$	$89.2^{90.8}_{87.6}$	$83.3^{84.9}_{81.4}$	$84.6^{86.6}_{82.6}$	$81.6^{83.2}_{80.1}$	$85.1^{87.1}_{83.0}$
Shapely, zeros	$81.0^{82.8}_{79.3}$	$84.9^{86.8}_{83.1}$	$82.0^{83.7}_{80.2}$	$82.4^{84.3}_{80.4}$	$81.6^{83.4}_{79.9}$	$83.2^{85.0}_{81.2}$
J_f^+ **(ours)**	$76.9^{78.7}_{74.9}$	$92.9^{94.5}_{91.5}$	$77.5^{79.4}_{75.5}$	$86.1^{88.3}_{83.8}$	$87.9^{89.5}_{86.3}$	$\mathbf{98.2^{98.9}_{97.4}}$

Table 1: Comparison of attribution methods (rows), and combinations of training/regularization schemes (columns). Our proposed method uses regularized hybrid contrastive learning. Numbers average across different configurations of number of factors (4 to 9), for 10 different datasets. Sub- and superscript values denote the 95% confidence interval obtained through bootstrapping (n=1,000).

Regularized, hybrid contrastive learning identifies the ground truth attribution map. Table 1 shows the AUC for recovering $\mathbf{A}$ using combinations of training schemes (supervised, supervised contrastive, hybrid contrastive), Jacobian regularization, and methods for estimating attribution methods. We investigate the effect of the different factors with an ordinary least squares (OLS) ANOVA (F=17.0, p<1e-5) followed by a Tukey HSD posthoc test, see Appendix B.4 for statistical methods and full results. Both the combination of regularized training followed by estimating the pseudo-inverse (p<0.01), and combining regularized training with hybrid contrastive learning (p<0.001) significantly outperform all considered baselines.

Contrastive learning is critical for large numbers of latent factors. The importance of using hybrid contrastive learning (which can identify the latent factors) becomes most apparent with an increasing number of latent factors, as we would expect in a realistic dataset. Fig 2 shows the variation in performance as we keep the number of observable factors fixed at 2 and vary the number of total latents from 4 to 9 variables. Beyond this value, the drop in R^2 becomes too large, prohibiting us to compute a meaningful attribution map. Performance scales with the number of available training samples, and we observed that increasing dataset size besides 100,000 samples allows to work with even higher numbers of latents.

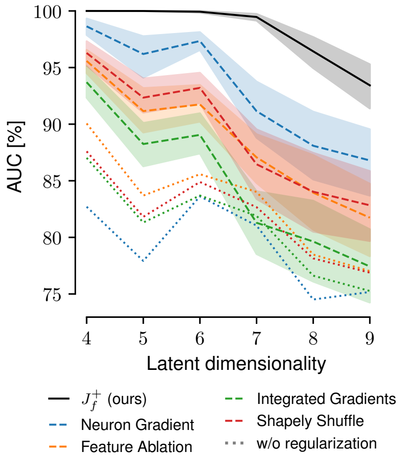


Figure 2: RegCL (ours, black) and supervised baselines AUC vs.# of latent factors.

Hybrid contrastive learning allows attribution computation with latent factors. In contrast to supervised algorithms, hybrid contrastive learning allows us to estimate the attribution map with respect to latent factors, i.e., we treat $\mathbf{z}_1$ as the observable, and $\mathbf{z}_2$ as the latent factor. With hybrid contrastive learning, we can continue to estimate the attribution map at AUC=99.2% (Table 2).

	CL, no reg.	**RegCL**
Neuron Gradient	$69.3^{72.4}_{66.0}$	$91.9^{95.3}_{88.3}$
Feature Ablation	$77.1^{80.4}_{73.8}$	$86.7^{89.9}_{83.1}$
Integrated Gradient	$77.5^{79.6}_{75.4}$	$86.8^{89.9}_{83.4}$
Shapely shuffled	$74.4^{77.5}_{71.2}$	$87.5^{91.0}_{84.0}$
Shapely, zeros	$75.8^{78.7}_{72.6}$	$85.3^{88.6}_{82.0}$
$\mathbf{J_f^+}$ **(ours)**	$84.2^{86.7}_{81.2}$	$\mathbf{99.2^{99.8}_{98.4}}$

Table 2: Contrastive learning (CL) can estimate attribution maps w.r.t. latent factors: Results for identifying the attribution map, avg. across 10 seeds and 4–9 latents.

4 Conclusions

We proposed a novel approach for estimating attribution maps in time-series data based on regularized, hybrid contrastive learning. Scientific inference in non-linear problems requires identifiable attribution maps estimated for the *data generating process*. We theoretically and empirically showed that contrastive learning can be leveraged to estimate this map by inverting the data generating process. Our empirical results demonstrate the importance of estimating *all latent variables* along with the observable factors for effective estimation of the attribution map. In future extensions of this work, we will apply our approach to real scientific data, e.g., for applications in neuroscience. Even beyond, we think that our findings might spark future work in improving the estimation of global explanations in vision, speech, and language.

Acknowledgements

We thank the anonymous reviewer for their helpful comments to improve the manuscript. Funding was provided by a Google PhD Fellowship to StS, NIH 1UF1NS126566-01 to MWM. StS acknowledges the IMPRS-IS Tübingen and ELLIS PhD program. MWM is the Bertarelli Foundation Chair of Integrative Neuroscience.

Author contributions

Conceptualization: StS, MWM; Methodology: StS, MWM, RG; Software: RG, StS; Theory: StS; Formal analysis: StS, RG, MF; Investigation: RG, StS, MF; Writing–Original Draft: StS, MWM; Writing–Editing: all authors.

References

[1] Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus H. Gross. Towards better understanding of gradient-based attribution methods for deep neural networks. In *International Conference on Learning Representations*, 2017. URL https://api.semanticscholar.org/CorpusID:3728967.

[2] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

[3] Osbert Bastani, Carolyn Kim, and Hamsa Bastani. Interpreting blackbox models via model extraction. *arXiv preprint arXiv:1705.08504*, 2017.

[4] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6541–6549, 2017.

[5] Richard Breen, Kristian Bernt Karlson, and Anders Holm. Interpreting and understanding logits, probits, and other nonlinear probability models. *annual review of sociology*, 44:39–54, 2018.

[6] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[7] Judy Hoffman, Daniel A Roberts, and Sho Yaida. Robust learning with jacobian regularization. *arXiv preprint arXiv:1908.02729*, 2019.

[8] Aapo Hyvarinen and Hiroshi Morioka. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. *Advances in neural information processing systems*, 29, 2016.

[9] Aapo Hyvarinen, Hiroaki Sasaki, and Richard Turner. Nonlinear ica using auxiliary variables and generalized contrastive learning. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 859–868. PMLR, 2019.

[10] Mehrdad Jazayeri and Srdjan Ostojic. Interpreting neural computations by examining intrinsic and embedding dimensionality of neural activity. *Current Opinion in Neurobiology*, 70:113–120, 2021.

[11] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pp. 2668–2677. PMLR, 2018.

[12] Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, et al. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*, 2020.

[13] Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Unmasking clever hans predictors and assessing what machines really learn. *Nature communications*, 10(1):1–8, 2019.

[14] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[15] Peter McCullagh and John A. Nelder. Generalized linear models. In *Predictive Analytics*, 1972. URL https://api.semanticscholar.org/CorpusID:14154576.

[16] Christoph Molnar. *Interpretable Machine Learning*. 2 edition, 2022. URL https://christophm.github.io/interpretable-ml-book.

[17] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognit.*, 65:211–222, 2015. URL https://api.semanticscholar.org/CorpusID:5731985.

[18] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern recognition*, 65:211–222, 2017.

[19] Hiroshi Morioka and Aapo Hyvarinen. Connectivity-contrastive learning: Combining causal discovery and representation learning for multimodal data. In *International Conference on Artificial Intelligence and Statistics*, pp. 3399–3426. PMLR, 2023.

[20] Pramod Kaushik Mudrakarta, Ankur Taly, Mukund Sundararajan, and Kedar Dhamdhere. Did the model understand the question?, 2018.

[21] Judea Pearl. *Causality*. Cambridge university press, 2009.

[22] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

[23] Patrik Reizinger, Yash Sharma, Matthias Bethge, Bernhard Schölkopf, Ferenc Huszár, and Wieland Brendel. Jacobian-based causal discovery with nonlinear ica. *Transactions on Machine Learning Research*, 2022.

[24] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

[25] Wojciech Samek, Grégoire Montavon, Andrea Vedaldi, Lars Kai Hansen, and Klaus-Robert Müller. *Explainable AI: interpreting, explaining and visualizing deep learning*, volume 11700. Springer Nature, 2019.

[26] Steffen Schneider, Jin Hwa Lee, and Mackenzie W. Mathis. Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, 617:360 – 368, 2023.

[27] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

[28] Lloyd S Shapley et al. A value for n-person games. 1953.

[29] Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences. *ArXiv*, abs/1605.01713, 2016. URL https://api.semanticscholar.org/CorpusID:8564234.

[30] Avanti Shrikumar, Jocelin Su, and Anshul Kundaje. Computationally efficient measures of internal neuron importance, 2018.

[31] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034, 2013.

[32] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[33] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, 2017. URL `https://api.semanticscholar.org/CorpusID:16747630`.

[34] Anne E. Urai, Brent Doiron, Andrew Michael Leifer, and Anne K. Churchland. Large-scale neural recordings call for new insights to link brain and behavior. *Nature Neuroscience*, 25: 11–19, 2022.

[35] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *ArXiv*, abs/1807.03748, 2018.

[36] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pp. 9929–9939. PMLR, 2020.

[37] Roland S Zimmermann, Yash Sharma, Steffen Schneider, Matthias Bethge, and Wieland Brendel. Contrastive learning inverts the data generating process. In *International Conference on Machine Learning*, pp. 12979–12990. PMLR, 2021.

A Proofs

We will now derive identifiability guarantees for the global attribution map under the ICA model described in the main paper. Given a data generating process and a ground truth global attribution map of the data generating process, we aim for a guarantee of the form

$$\hat{\mathbf{J}}_{\mathbf{g}} = \mathbf{J}_{\mathbf{g}} \odot \mathbf{L} \tag{5}$$

for a suitable estimator $\hat{\mathbf{J}}_{\mathbf{g}}$ up to a matrix $\mathbf{L}$ that scales the ground truth derivatives in $\mathbf{J}_{\mathbf{g}}$ point-wise and will hence not affect the "real zeros" in the Jacobians relevant for Def. 2.

We use contrastive learning to obtain a feature encoder $\mathbf{f}$ which identifies the ground-truth latents up to a linear indeterminacy. We structure this feature encoder to reconstruct different parts of the latent representation in different dimensions of the reconstructed latent space.

Then, we estimate the attribution map by computing the pseudo-inverse of the feature encoder's Jacobian, which is directly related to the Jacobian of the mixing function. To obtain the correct pseudo-inverse, we need to obtain a minimum-Jacobian solution of the feature encoding network. We hence introduce a new regularized contrastive learning objective.

The underlying constrained optimization problem is

$$\min_{\mathbf{f}} \|\mathbf{J}_{\mathbf{f}}(\mathbf{x})\|_F^2 \quad \text{s.t.} \quad \phi_i(\mathbf{f}_i(\mathbf{x}), \mathbf{f}_i(\mathbf{y})) = \log \frac{p_i(\mathbf{y}|\mathbf{x})}{q(\mathbf{y}|\mathbf{x})} + C_i(\mathbf{x}) \quad \forall i \in [G], \tag{6}$$

with the positive sample distribution p_i and the negative sample distribution q. We call $(\mathbf{x}, \mathbf{y}_+)$ the positive pair, and all $(\mathbf{x}, \mathbf{y}_i^-)$ negative pairs. In the following we define $\psi_i(\mathbf{x}, \mathbf{y}) := \phi_i(\mathbf{f}_i(\mathbf{x}), \mathbf{f}_i(\mathbf{y}))$ where $\mathbf{f} := [\mathbf{f}_1; \ldots; \mathbf{f}_G]$ is the feature encoder and ϕ_i are similarity metrics. We re-state the RegCL objective function which is a relaxation of Eq. 6:

$$\mathcal{L}_N[\psi; \lambda] = \mathop{\mathbb{E}}_{\substack{\mathbf{x} \sim p(\mathbf{x}), \\ \mathbf{y}^+ \sim p_i(\mathbf{y}|\mathbf{x}) \; \forall i \in [G] \\ \mathbf{y}_1^- \cdots \mathbf{y}_N^- \sim q(\mathbf{y}|\mathbf{x})}} \left[\sum_{i=1}^{G} \Big(-\psi_i(\mathbf{x}, \mathbf{y}_i^+) + \log \sum_{j=1}^{N} e^{\psi_i(\mathbf{x}, \mathbf{y}_j^-)} \Big) + \lambda \|\mathbf{J}_{\mathbf{f}}(\mathbf{x})\|_F^2 \right]. \tag{7}$$

In principle, this objective is able to identify an arbitrary amount of separate factor groups (G), given sufficient capacity of the model. The choice of ψ_i for the individual parts of the feature representation depends on the exact distribution underlying data generation, and is discussed below.

A.1 Preliminaries

Before proving our results on identifiable attribution maps, it is useful to restate a few known results from the literature, concerning properties of the InfoNCE loss. Hyvarinen et al. [9] showed that contrastive learning with auxiliary variables is identifiable up to permutations or linear transformations for condtionally expontential distributions. Zimmermann et al. [37] related this to identifiability for models trained with the InfoNCE loss, and showed that assumptions about the data-generating process can be incorporated in to the choice of loss function. Schneider et al. [26] then formulated a supervised contrastive learning objective based on selecting the positive and negative distributions in the generalized InfoNCE objective.

We will first re-state the minimizer of the InfoNCE loss 2 used in our algorithm:

Proposition 1 (restated from Schneider et al. [26])**.** *Let $p(\cdot|\cdot)$ be the conditional distribution of the positive samples, $q(\cdot|\cdot)$ the conditional distribution of the negative samples and $p(\cdot)$ the marginal distribution of the reference samples. The generalized InfoNCE objective (Def. 2) is convex in ψ with the unique minimizer*

$$\psi^*(\mathbf{x}, \mathbf{y}) = \log \frac{p(\mathbf{y}|\mathbf{x})}{q(\mathbf{y}|\mathbf{x})} + C(\mathbf{x}), \quad \textit{with} \quad \mathcal{L}_N[\psi^*] = \log N - \mathcal{D}_{\mathrm{KL}}(p(\cdot|\cdot)\|q(\cdot|\cdot)) \tag{8}$$

for $N \to \infty$ on the support of $p(\mathbf{x})$, where $C : \mathbb{R}^d \to \mathbb{R}$ is an arbitrary mapping.

Proof. See [26], but note that we added the batch size N. □

We also re-state

Proposition 2 (restated Proposition 6 in Schneider et al. [26]). *Assume the learning setup in Def. 1 [26], and that the ground-truth latents $\mathbf{u}_1, \dots, \mathbf{u}_T$ for each time point follow a uniform marginal distribution and the change between subsequent time steps is given by the conditional distribution of the form*

$$p(\mathbf{u}_{t+\Delta t}|\mathbf{u}_t) = \frac{1}{Z(\mathbf{u}_t)} \exp \delta(\mathbf{u}_{t+\Delta t}, \mathbf{u}_t) \tag{9}$$

where δ is either a (scaled) dot product (and $\mathbf{u}_t \in \mathcal{S}^{n-1} \subset \mathbb{R}^d$ lies on the $(n-1)$-sphere $\mathcal{S}^{n-1}$) or an arbitrary semi-metric (and $\mathbf{u}_t \in \mathcal{U} \subset \mathbb{R}^d$ lies in a convex body $\mathcal{U}$). Assume that the data generating process $\mathbf{g}$ with $\mathbf{s}_t = \mathbf{g}(\mathbf{u}_t)$ is injective. Assume we train a symmetric CEBRA [26] model with encoder $\mathbf{f} = \mathbf{f}'$ and the similarity measure including a fixed temperature $\tau > 0$ is set to or sufficiently flexible such that $\phi = \delta$ for all arguments. Then $\mathbf{h} = \mathbf{h}' = \mathbf{g} \circ \mathbf{f}$ is affine.

Proof. For δ being the dot product, the result follows from the proof of Theorem 2 in Zimmermann et al. [37]. For δ being a semi-metric, the result follows from the proof of Theorem 5 in Zimmermann et al. [37]. □

A.2 Positive distributions for self-supervised and supervised contrastive learning

Self-supervised contrastive learning Up to one of the parts in the latent representation $\mathbf{z}$ can be estimated using self-supervised learning by leveraging time information in the signal. The underlying assumption is that latents vary over time according to a distribution we can model with ψ. For instance, Brownian motion $p(\mathbf{z}^{(t+1)}|\mathbf{z}^{(t)}) = \mathcal{N}(\mathbf{z}^{(t+1)} - \mathbf{z}^{(t)}|0, \sigma^2\mathbf{I})$ can be estimated by selecting $\phi(\mathbf{x}, \mathbf{y}) = -\|\mathbf{x} - \mathbf{y}\|^2$. On the hypersphere with a vMF conditional across timesteps, the dot product is a suitable choice for $\phi(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\top \mathbf{y}$. Due to Proposition 2, this training scheme is able to identify the ground truth latents up to a linear inderterminancy.

Supervised contrastive learning For supervised contrastive learning, we uniformly sample a timestep (and hence, a sample $\mathbf{x}$) from the dataset. This timestep is associated to the label $\mathbf{c}$, and we then sample $\mathbf{c}'$ from the conditional distribution $p(\mathbf{c}'|\mathbf{c})$. We select the nearest neighbour to $\mathbf{c}'$ with the corresponding sample $\mathbf{x}'$.

The conditional distribution $p(\mathbf{c}'|\mathbf{c})$ can be contructed as an *empirical* distribution: For instance, if we assume non-stationarity, $\mathbf{c}^{(t-1)} - \mathbf{c}^{(t)}$ can be computed across the dataset. Let us call this distribution $\hat{p}(\mathbf{c}' - \mathbf{c})$. Then, sampling from $p(\mathbf{c}'|\mathbf{c})$ can take the form of sampling $\mathbf{c}' = \mathbf{c} + \Delta$ with $\Delta \sim \hat{p}(\mathbf{c}' - \mathbf{c})$.

If this approximation is correct under the underlying latent distribution, have we have $p(\mathbf{c}'|\mathbf{c}) \det \mathbf{J}_\gamma^{-1}(\mathbf{c}') = p(\mathbf{z}'|\mathbf{z})$. This means that the solutions of the supervised and self-supervised contrastive learning solutions coincide.

Superposition of self-supervised and supervised contrastive learning Depending on the assumptions about the ground truth data distribution, different estimation schemes can be combined to obtain a latent representation. In the end, the feature encoder $\mathbf{f}$ should identify the original latents $\mathbf{z}$ up to a linear transformation,

$$\mathbf{f}(\mathbf{g}(\mathbf{z})) = \mathbf{L}\mathbf{z}. \tag{10}$$

Our goal is to obtain block-structure in $\mathbf{L}$, with zeros in the lower block triangular part of the matrix.

This is possible by simultaneously solving multiple contrastive learning objectives, which requires

$$\mathbf{f}_i(\mathbf{g}(\mathbf{z})) = \mathbf{L}_i\mathbf{z}. \tag{11}$$

for each part i of the latent representation. Assume without loss of generality that we apply self-supervised contrastive learning to the G-th part, and supervised contrastive learning to all remaining parts. For supervised contrastive learning we then obtain

$$\mathbf{f}_i(\mathbf{g}(\mathbf{z})) = \mathbf{L}_i\mathbf{z} = \mathbf{L}_i'\mathbf{z}_i. \tag{12}$$

If all latents $\mathbf{z}$ satisfy the conditions for time-contrastive learning, we can then also apply time-contrastive learning to the full representation, which gives us the following constraints:

$$\mathbf{f}_i(\mathbf{g}(\mathbf{z})) = \mathbf{L}_i\mathbf{z} = \mathbf{L}_i'\mathbf{z}_i \quad \forall i \in [G-1] \tag{13}$$

$$\mathbf{f}(\mathbf{g}(\mathbf{z})) = \mathbf{L}\mathbf{z} \tag{14}$$

from which we can follow the matrix structure

$$\mathbf{f}(\mathbf{g}(\mathbf{z})) = \mathrm{diag}(\mathbf{L}_1, \ldots, \mathbf{L}_G) \tag{15}$$

In cases where this is not possible, note that it is always possible to treat all contrastive learning problems separately, and learn separate regions of the feature space in $\mathbf{f}$. This gives the same result, but re-uses less of the representation (e.g., the self-supervised part of the representation would be learned separately from the supervised part).

Consider a time-series dataset where $p(\mathbf{z}_t|\mathbf{z}_{t-1})$, i.e., all latents, follow Brownian motion. We can the produce the solution

$$\psi_i(\mathbf{x}, \mathbf{x}') := \phi_i(\mathbf{f}_i(\mathbf{x}), \mathbf{f}_i(\mathbf{x}')) = \log \frac{p(\mathbf{c}_i'|\mathbf{c}_i)}{q(\mathbf{c}_i'|\mathbf{c}_i)} \quad i \in \{1, \ldots, G-1\} \tag{16}$$

$$\psi_G(\mathbf{x}, \mathbf{x}') := \sum_{i=1}^{G} \phi_i(\mathbf{f}_i(\mathbf{x}), \mathbf{f}_i(\mathbf{x}')) = \log \frac{p(\mathbf{z}'|\mathbf{z})}{q(\mathbf{z}'|\mathbf{z})} = \log \frac{p(\mathbf{z}_G'|\mathbf{z}_G)}{q(\mathbf{z}_G'|\mathbf{z}_G)} + \sum_{i=1}^{G-1} \log \frac{p(\mathbf{c}_i'|\mathbf{c}_i)|\mathbf{J}_{\gamma_i}^{-1}(\mathbf{z}_i')|}{q(\mathbf{c}_i'|\mathbf{c}_i)|\mathbf{J}_{\gamma_i}^{-1}(\mathbf{z}_i')|} \tag{17}$$

in case our training distributions for supervised contrastive learning, $p(\mathbf{c}_i|\mathbf{c}_i)$ are a sufficiently good approximation of the variation in the ground truth latents, we can select $\psi_G(\mathbf{x}, \mathbf{y}) := \phi(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{y}))$ to be trained on the whole feature space using self-supervised learning, while all other objectives on ψ_i would solve supervised contrastive losses. If this training setup is not possible, it would be required to parametrize $\psi_G(\mathbf{x}, \mathbf{y}) := \phi(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{y}))$ as a separate part of the feature space.

While it is beyond the scope of the current work to thoroughly investigate the trade-offs between the two methods, our verification experiments assume the former case: The time contrastive objective is applied to the whole objective function, and the behavior contrastive objective to the previous latent variable groups.

A.3 Proof of Theorem 1

An interesting property of contrastive learning algorithms is the natural definition of a "goodness of fit" metric for the model. This goodness of fit can be derived from the value of the InfoNCE metric which is bounded from below and above as follows [26]:

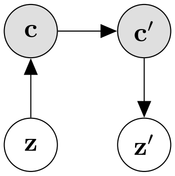
$$\log N - D_{\mathrm{KL}}(p||q) \leq \mathcal{L}_N[\psi] \leq \log N. \tag{18}$$

In scientific applications, we can leverage the distance to the trivial solution $\log N$ as a quality measure for the model fit. Theorem 1 states that if during supervised contrastive learning with labels $\mathbf{c}$ there is no meaningful relation between $\mathbf{c}$ and $\mathbf{x}$, we will observe a trivial solution with loss value at $\log N$.

For the following proof, let us recall from Def. 1 that we can split the latents $\mathbf{z}$ that fully define the data through the mixing function, $\mathbf{x} = \mathbf{g}(\mathbf{z})$. We can split $\mathbf{z}$ into different parts, $\mathbf{z} = [\mathbf{z}_1, \ldots, \mathbf{z}_G]$ and assume that $\mathbf{c}_i$ is the observable factor corresponding to the i-th part. For notational brevity, we omit the i in the following formulation of the proof without loss of generality.

Proof of Theorem 1

Proof. Assume that the distribution p is informed by labels. In the most general case, we can depict the sampling scheme for supervised contrastive learning with continuous labels $\mathbf{c}$ and $\mathbf{c}'$ and latents $\mathbf{z}$ and $\mathbf{z}'$ with the following graphical model:



The reference sample $\mathbf{x}$ is linked to the observable factor/label $\mathbf{c}$, and the conditional $p(\mathbf{c}'|\mathbf{c})$ links both samples. In particular, $\mathbf{z}'$ and hence $\mathbf{x}'$ are selected based on $\mathbf{c}'$ in the dataset.

The distributions for positive and negative samples then factorize into

$$p(\mathbf{z}'|\mathbf{z}) = \int\int d\mathbf{c}' d\mathbf{c} p(\mathbf{z}'|\mathbf{c}')p(\mathbf{c}'|\mathbf{c})p(\mathbf{c}|\mathbf{z}) \tag{19}$$

$$q(\mathbf{z}'|\mathbf{z}) = \int\int d\mathbf{c}' d\mathbf{c} p(\mathbf{z}'|\mathbf{c}')q(\mathbf{c}'|\mathbf{c})p(\mathbf{c}|\mathbf{z}) \tag{20}$$

and note that only $p(\mathbf{c}'|\mathbf{c})$ and $q(\mathbf{c}'|\mathbf{c})$ are selected by the user of the algorithm, the remaining distributions are empirical properties of the dataset.

We can compute the density ratio

$$\frac{p(\mathbf{z}'|\mathbf{z})}{q(\mathbf{z}'|\mathbf{z})} = \frac{\int\int d\mathbf{c}' d\mathbf{c} p(\mathbf{z}'|\mathbf{c}')p(\mathbf{c}'|\mathbf{c})p(\mathbf{c}|\mathbf{z})}{\int\int d\mathbf{c}' d\mathbf{c} p(\mathbf{z}'|\mathbf{c}')q(\mathbf{c}'|\mathbf{c})p(\mathbf{c}|\mathbf{z})} \tag{21}$$

In the case where latents and observables are independent variables, we have $p(\mathbf{z}'|\mathbf{c}') = p(\mathbf{z}')$ and $p(\mathbf{c}|\mathbf{z}) = p(\mathbf{c})$. The equation then reduces to

$$= \frac{\int\int d\mathbf{c}' d\mathbf{c} p(\mathbf{z}')p(\mathbf{c}'|\mathbf{c})p(\mathbf{c})}{\int\int d\mathbf{c}' d\mathbf{c} p(\mathbf{z}')q(\mathbf{c}'|\mathbf{c})p(\mathbf{c})} \tag{22}$$

$$= \frac{p(\mathbf{z}')\int\int d\mathbf{c}' d\mathbf{c} p(\mathbf{c}'|\mathbf{c})p(\mathbf{c})}{p(\mathbf{z}')\int\int d\mathbf{c}' d\mathbf{c} q(\mathbf{c}'|\mathbf{c})p(\mathbf{c})} = 1. \tag{23}$$

Consequently, the minimizer is $\psi(\mathbf{x}, \mathbf{y}) = C(\mathbf{x})$ and we obtain the maximum value of the loss with $\mathcal{L}[\psi] = \log N$ in the limit of $N \to \infty$. Note, for any symmetrically parametrized similarity metric (like the cosine or Euclidean loss), it follows that $\psi(\mathbf{x}, \mathbf{y}) = \psi$ is constant, i.e., the function collapses onto a single point.

□

A.4 Proof of Theorem 2

Proof. If $\mathbf{f}$ is a minimizer of the InfoNCE loss under the assumed generative model, it follows that we part-wise identify the underlying latents,

$$\mathbf{f}(\mathbf{g}(\mathbf{z})) = \mathbf{B}\mathbf{z} \tag{24}$$

with some block diagonal matrix $\mathbf{B}$. By taking the derivative w.r.t. $\mathbf{z}$ it follows that

$$\mathbf{J}_\mathbf{f}(\mathbf{x})\mathbf{J}_\mathbf{g}(\mathbf{z}) = \mathbf{B}. \tag{25}$$

We need to show that at each point $\mathbf{z}$ in the factor space, we can recover $\mathbf{J}_g$ up to some indeterminacy. We will re-arrange the equation to obtain

$$\mathbf{J}_\mathbf{f}(\mathbf{x})\mathbf{J}_\mathbf{g}(\mathbf{z})\mathbf{B}^{-1} = \mathbf{I}, \tag{26}$$

$$\mathbf{J}_\mathbf{f}(\mathbf{x})\tilde{\mathbf{J}}_\mathbf{g}(\mathbf{z}) = \mathbf{I}. \tag{27}$$

It is clear that for each point in the support of p, $\mathbf{J}_\mathbf{f}(\mathbf{x})$ is a left inverse of $\tilde{\mathbf{J}}_\mathbf{g}(\mathbf{z})$.

$$\mathbf{J}_\mathbf{f}(\mathbf{x}) = \tilde{\mathbf{J}}^+_\mathbf{g}(\mathbf{z}) + \mathbf{V}, \mathbf{v}_i \in \ker \tilde{\mathbf{J}}_\mathbf{g}(\mathbf{z}) \tag{28}$$

Among these solutions, it is well-known that the minimum norm solution $\mathbf{J}^*$ to

$$\min_{\mathbf{J}(\mathbf{z})} \|\mathbf{J}(\mathbf{z})\|_F^2 \text{ s.t. } \mathbf{J}(\mathbf{z})\mathbf{J}_\mathbf{g}(\mathbf{z}) = \mathbf{I} \tag{29}$$

is the Moore-Penrose inverse, $\mathbf{J}^*(\mathbf{z}) = \tilde{\mathbf{J}}^+_\mathbf{g}(\mathbf{z})$. By invoking assumption (2), we arrive at this solution and have

$$\mathbf{J}_\mathbf{f}(\mathbf{x}) = \tilde{\mathbf{J}}^+_\mathbf{g}(\mathbf{z}) \tag{30}$$

$$\mathbf{J}^+_\mathbf{f}(\mathbf{x}) = \tilde{\mathbf{J}}_\mathbf{g}(\mathbf{z}) \tag{31}$$

$$\mathbf{J}^+_\mathbf{f}(\mathbf{x}) = \mathbf{J}_\mathbf{g}(\mathbf{z})\mathbf{B}^{-1} \tag{32}$$

Because $\mathbf{B}$ is block-diagonal with zeros in the off-diagonal blocks, this also applies to $\mathbf{B}^{-1}$. It follows that

$$\mathbf{J}^+_\mathbf{f}(\mathbf{x}) = \mathbf{J}^+_\mathbf{f}(\mathbf{g}(\mathbf{z})) \propto \mathbf{J}_\mathbf{g}(\mathbf{z}) \tag{33}$$

concluding the proof. □

B Implementation notes

B.1 Obtaining the attribution map

Since $\mathbf{J}_\mathbf{f}^+$ identifies $\mathbf{J}_\mathbf{g}$ as derived in Theorem 2, we can obtain the final attribution map according to Def. 2 using

$$\hat{\mathbf{A}} = \mathbf{1}\{\max_{\mathbf{x}} |\mathbf{J}_\mathbf{f}^+(\mathbf{x})| > \epsilon\} \tag{34}$$

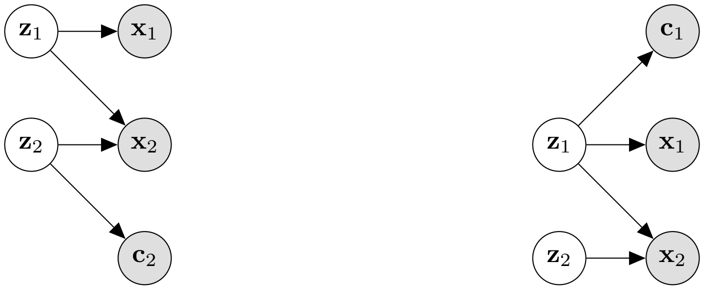
where $\epsilon > 0$ is a threshold that weights false-positive and false-negative predictions. In practice, we found that the operation

$$\hat{\mathbf{A}} = \mathbf{1}\{\sum_{\mathbf{x}} |\mathbf{J}_\mathbf{f}^+(\mathbf{x})|\} > \epsilon \tag{35}$$

yields even better performance, and we will use this estimation method for all experiments. In general, working on improved estimation methods taking into account sources of estimation noise could be an interesting avenue for future work.

B.2 Synthetic data design

An essential aspect of our synthetic design lies in the definition of the mixing function $\mathbf{g}$ which, consequently, defines the ground truth attribution map. We split the factors $\mathbf{z}$ into two parts, $\mathbf{z}_1$ and $\mathbf{z}_2$. Figure B.2 illustrates the two experimental configurations employed in this work. In both settings $\mathbf{z}_1$ is connected both to $\mathbf{x}_1$ and $\mathbf{x}_2$ whereas $\mathbf{z}_2$ is only be connected to $\mathbf{x}_2$. The main difference is that in the first setting $\mathbf{z}_2 = \gamma_2(\mathbf{c}_2)$ whereas in the second setting $\mathbf{z}_1 = \gamma_1(\mathbf{c}_1)$.



(a) Graphical model for the data generating process where $\mathbf{z}_2$ is observed through $\mathbf{c}_2$. The attribution map needs to be computed with respect to $\mathbf{z}_2$, which is inferred with supervised (contrastive) learning. This is the experiment setting for Table 1.

(b) Graphical model for the data generating process where $\mathbf{z}_1$ is observed through $\mathbf{c}_1$. Since $\mathbf{z}_2$ is not observed, the attribution map can only be estimated through the time-contrastive component in RegCL. This is the experiment setting for Table 2.

B.3 Detailed experimental setup

In our experiments, we consider variations of three factors. Our theory predicts that the combination of estimating the inverse of the feature encoder Jacobian with regularized training allows to identify the ground truth attribution map. We test the following factors and underline our proposed method:

Factor	Possible values
Training mode	Supervised, Supervised contrastive, <u>Hybrid contrastive</u>
Regularization	Off, <u>On ($\lambda = 0.1$)</u>
Attribution map estimation	Neuron gradient, integrated gradients, Shapely values, <u>inverted Jacobian</u>

Combinations of these factors can have positive effects on the output performance. We therefore run all combinations of these factors with 10 seeds (i.e., different latents and mixing functions) across different numbers of latent dimensions.

B.4 Statistical analysis

We fit an ANOVA on an ordinary least squares model using combinations of all latent factors, see Table 3. As a post-hoc test, we use a Tukey HSD test on the statistically significant factors. See Table 4 we show that hybrid contrastive learning computing followed by computing the pseudo-inverse significantly outperforms all other methods, and in Table 5 we show that combining the pseudo-inverse on regularized trained models also significantly outperform all other methods. Statistical analysis is implemented using `statsmodels`[2].

	sum sq	df	F	PR(>F)
C(attribution method name)	807.50	5	6.14	0.00
C(dim Z1)	3286.82	5	24.98	0.00
C(method name)	1505.46	2	28.60	0.00
C(extension)	15722.40	1	597.37	0.00
C(attribution method name):C(dim Z1)	456.86	25	0.69	0.85
C(attribution method name):C(method name)	8747.26	10	33.24	0.00
C(dim Z1):C(method name)	270.05	10	1.03	0.41
C(attribution method name):C(extension)	6661.36	5	50.62	0.00
C(dim Z1):C(extension)	2647.68	5	20.12	0.00
C(method name):C(extension)	2813.94	2	53.46	0.00
C(attribution method name):C(dim Z1):C(method name)	463.75	50	0.35	1.00
C(attribution method name):C(dim Z1):C(extension)	672.62	25	1.02	0.43
C(attribution method name):C(method name):C(extension)	177.68	10	0.68	0.71
C(dim Z1):C(method name):C(extension)	237.40	10	0.90	0.51
C(attribution method name):C(dim Z1):C(method name):C(extension)	932.86	50	0.71	0.93
Residual	50059.50	1902	NaN	NaN

Table 3: Results for fitting an ANOVA on all combination of factors.

group1	group2	meandiff	p-adj	lower	upper	reject
$J_{\mathbf{f}}^{+}$:hybrid contrastive	$J_{\mathbf{f}}$:behavior contrastive	9.41	0.00	5.83	12.98	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	$J_{\mathbf{f}}$:hybrid contrastive	9.51	0.00	5.93	13.08	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	$J_{\mathbf{f}}$:supervised	6.97	0.00	3.40	10.55	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	$J_{\mathbf{f}}^{+}$:behavior contrastive	11.29	0.00	7.71	14.87	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	integrated-gradients:hybrid contrastive	-7.67	0.00	-11.75	-3.59	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	feature-ablation:supervised	-7.23	0.00	-10.81	-3.66	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	feature-ablation:hybrid contrastive	-9.00	0.00	-12.58	-5.42	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	feature-ablation:behavior contrastive	-8.74	0.00	-12.32	-5.16	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	$J_{\mathbf{f}}^{+}$:supervised	-8.14	0.00	-11.72	-4.57	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	shapely-zeros:hybrid contrastive	-10.68	0.00	-14.26	-7.10	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	shapely-shuffle:hybrid contrastive	-9.71	0.00	-13.29	-6.13	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	shapely-shuffle:supervised	-7.48	0.00	-11.06	-3.90	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	shapely-zeros:behavior contrastive	-10.90	0.00	-14.47	-7.32	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	shapely-zeros:supervised	-10.09	0.00	-13.66	-6.51	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	integrated-gradients:behavior contrastive	-10.96	0.00	-14.54	-7.38	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	integrated-gradients:supervised	-10.14	0.00	-13.72	-6.57	True
$J_{\mathbf{f}}^{+}$:hybrid contrastive	shapely-shuffle:behavior contrastive	-9.13	0.00	-12.71	-5.55	True
$J_{\mathbf{f}}$:supervised	$J_{\mathbf{f}}^{+}$:behavior contrastive	-4.31	0.00	-7.89	-0.74	True
$J_{\mathbf{f}}$:supervised	shapely-zeros:hybrid contrastive	-3.70	0.03	-7.28	-0.13	True
$J_{\mathbf{f}}$:supervised	shapely-zeros:behavior contrastive	-3.92	0.02	-7.50	-0.35	True
$J_{\mathbf{f}}$:supervised	integrated-gradients:behavior contrastive	-3.99	0.01	-7.56	-0.41	True
feature-ablation:supervised	$J_{\mathbf{f}}^{+}$:behavior contrastive	4.05	0.01	0.48	7.63	True
feature-ablation:supervised	integrated-gradients:behavior contrastive	-3.73	0.03	-7.30	-0.15	True
feature-ablation:supervised	shapely-zeros:behavior contrastive	-3.66	0.04	-7.24	-0.09	True
shapely-shuffle:supervised	$J_{\mathbf{f}}^{+}$:behavior contrastive	3.81	0.02	0.23	7.39	True

Table 4: Post-hoc test for the combination of attribution method and training method.

[2] https://github.com/statsmodels/statsmodels/

group1	group2	meandiff	p-adj	lower	upper	reject
$J_{\mathbf{f}}^{+}$:REG	$J_{\mathbf{f}}$:REG	3.16	0.00	0.58	5.75	True
$J_{\mathbf{f}}^{+}$:REG	shapely-zeros:REG	-8.91	0.00	-11.50	-6.32	True
$J_{\mathbf{f}}^{+}$:REG	$J_{\mathbf{f}}^{+}$:none	-11.61	0.00	-14.20	-9.03	True
$J_{\mathbf{f}}^{+}$:REG	feature-ablation:REG	-6.25	0.00	-8.84	-3.67	True
$J_{\mathbf{f}}^{+}$:REG	feature-ablation:none	-9.06	0.00	-11.64	-6.47	True
$J_{\mathbf{f}}^{+}$:REG	integrated-gradients:REG	-8.02	0.00	-10.70	-5.35	True
$J_{\mathbf{f}}^{+}$:REG	integrated-gradients:none	-10.38	0.00	-13.06	-7.70	True
$J_{\mathbf{f}}^{+}$:REG	shapely-shuffle:REG	-6.11	0.00	-8.69	-3.52	True
$J_{\mathbf{f}}^{+}$:REG	shapely-shuffle:none	-10.10	0.00	-12.69	-7.51	True
$J_{\mathbf{f}}^{+}$:REG	$J_{\mathbf{f}}$:none	12.75	0.00	10.17	15.34	True
$J_{\mathbf{f}}^{+}$:REG	shapely-zeros:none	-10.86	0.00	-13.44	-8.27	True
$J_{\mathbf{f}}$:REG	shapely-zeros:REG	-5.75	0.00	-8.33	-3.16	True
$J_{\mathbf{f}}$:REG	shapely-shuffle:REG	-2.94	0.01	-5.53	-0.35	True
$J_{\mathbf{f}}$:REG	integrated-gradients:none	-7.21	0.00	-9.89	-4.53	True
$J_{\mathbf{f}}$:REG	integrated-gradients:REG	-4.86	0.00	-7.53	-2.18	True
$J_{\mathbf{f}}$:REG	feature-ablation:none	-5.89	0.00	-8.48	-3.30	True
$J_{\mathbf{f}}$:REG	feature-ablation:REG	-3.09	0.01	-5.68	-0.50	True
$J_{\mathbf{f}}$:REG	$J_{\mathbf{f}}^{+}$:none	-8.45	0.00	-11.04	-5.86	True
$J_{\mathbf{f}}$:REG	$J_{\mathbf{f}}$:none	-9.59	0.00	-12.18	-7.00	True
$J_{\mathbf{f}}$:REG	shapely-shuffle:none	-6.93	0.00	-9.52	-4.35	True
$J_{\mathbf{f}}$:REG	shapely-zeros:none	-7.69	0.00	-10.28	-5.10	True
shapely-shuffle:REG	$J_{\mathbf{f}}$:none	6.65	0.00	4.06	9.24	True
shapely-shuffle:REG	shapely-zeros:none	-4.75	0.00	-7.34	-2.16	True
shapely-shuffle:REG	shapely-zeros:REG	-2.81	0.02	-5.39	-0.22	True
shapely-shuffle:REG	$J_{\mathbf{f}}^{+}$:none	5.51	0.00	2.92	8.10	True
shapely-shuffle:REG	integrated-gradients:none	4.27	0.00	1.59	6.95	True
shapely-shuffle:REG	feature-ablation:none	2.95	0.01	0.36	5.54	True
shapely-shuffle:REG	shapely-shuffle:none	-3.99	0.00	-6.58	-1.41	True
feature-ablation:REG	feature-ablation:none	-2.80	0.02	-5.39	-0.21	True
feature-ablation:REG	shapely-shuffle:none	-3.84	0.00	-6.43	-1.26	True
feature-ablation:REG	$J_{\mathbf{f}}$:none	6.50	0.00	3.91	9.09	True
feature-ablation:REG	shapely-zeros:REG	-2.66	0.04	-5.24	-0.07	True
feature-ablation:REG	integrated-gradients:none	-4.12	0.00	-6.80	-1.44	True
feature-ablation:REG	$J_{\mathbf{f}}^{+}$:none	5.36	0.00	2.77	7.95	True
feature-ablation:REG	shapely-zeros:none	-4.60	0.00	-7.19	-2.01	True
integrated-gradients:REG	$J_{\mathbf{f}}^{+}$:none	3.59	0.00	0.92	6.27	True
integrated-gradients:REG	shapely-zeros:none	-2.83	0.03	-5.51	-0.16	True
integrated-gradients:REG	$J_{\mathbf{f}}$:none	4.73	0.00	2.06	7.41	True
shapely-zeros:REG	$J_{\mathbf{f}}$:none	3.84	0.00	1.26	6.43	True
shapely-zeros:REG	$J_{\mathbf{f}}^{+}$:none	2.70	0.03	0.11	5.29	True
feature-ablation:none	$J_{\mathbf{f}}$:none	3.70	0.00	1.11	6.29	True
shapely-shuffle:none	$J_{\mathbf{f}}$:none	2.66	0.04	0.07	5.24	True

Table 5: Posthoc test for the combination of attribution method and regularization scheme.

C Related Work

There are two main approaches to model understanding. The first approach is to use interpretable models from the start, e.g., linear regression. The second approach is to explain complex models using post-hoc interpretability methods. Unfortunately, the first approach is often not feasible due to complex non-linearities in the data, and therefore we focus on the second approach, making use of methods that will be discussed below, such as saliency maps.

Depending on the type of explanation we want to obtain, there are different post-hoc interpretability methods available in the literature [25]. First, we can differentiate between local and global explanations. *Global explanations* provide an interpretable description of the behavior of the model as a whole. *Local explanations* provide a description of the model behavior in a specific neighborhood/for an individual prediction.

In the case of local explanations, we can categorize the methods (non-extensively) in the following way:

Feature attribution methods are explanations where we assign a weight to each feature in the input space that indicates its importance or effect. We can distinguish between:

- *Perturbation based* compute a relevance score by removing, masking or altering the input, running a forward pass on the new input and measuring the difference with the original input. Methods include LIME or SHAP [24, 14].
- *Gradient based* methods locally evaluate the gradient $\partial f/\partial x_i$ or variations of it (e.g., absolute value of the gradient). Methods include Integrated Gradients, SmoothGrad, or Grad-CAM [33, 32, 27].
- *Propagation based* methods decompose the prediction of the network going backward (from output to input) following some propagation rules. Common methods are Deep Taylor decomposition and Layer Relevance Propagation (LRP) [18, 2].

Prototype-based methods are methods based on creating a prototype in the input domain that is interpretable and representative of the abstract learned concept, such as activation maximization [35]. This is used to answer questions such as: What type of input is easier to mis-classify?

For *global explanations* we can differentiate between:

- *Meta-explanations methods* aggregate and analyze a collection of multiple individual explanations to identify general patterns in the model behavior. A recent example is SpRAy [13], which computes meta-explanations by clustering individual heatmaps.
- *Representation-based methods* analyze intermediate representations of a neural network. An example of this approach is network dissection [4], which consists of evaluating the semantics of hidden units to determine the model's reliance on concepts that are semantically similar to humans. Another example is TCAV [11], which measures the sensitivity of a model's prediction in terms of user-defined concepts.
- *Model distillation methods* create a simpler and more interpretable model that is constructed such that it mimics the original model's predictions. An example is using decision trees [3].

Causal discovery and Identifiability The goal of causal discovery is to learn the causal structure of the data, often represented as a Directed Acyclic Graph (DAG) [21, 22]. Importantly, there is a deep connection between causal discovery and identifiability as both aim to infer the ground truth data generating process. As a result, a growing number of studies are showcasing this connection [19, 23].

D Additional Discussion and Limitations

We demonstrated a theoretically grounded algorithm for estimating attribution maps with identifiability guarantees. While we were able to demonstrate its performance on synthetic datasets matching the theoretical conditions up to real-world data.

Our theoretical results currently hold for fully converged contrastive learning models and true minimizers of the InfoNCE loss [35] in the limit of infinite data. While Wang & Isola [36] show favorable properties of contrastive learning in limited data settings, which can be confirmed by our finite data experiments, it is less straightforward to theoretically connect the quality of the attribution score to the goodness of fit of the model. For the purpose of this work, we show that the R^2 of recovering the observable factors is a good indicator, and recommend comparing this to the theoretically best result (of a supervised baseline).

In future work, the presented results should be extended and studied under various violations of the data-distributions, and scaled to real-world datasets.