A robust estimator of mutual information for deep learning interpretability

Davide Piras*
Department of Physics & Astronomy
University College London
Gower Street, London WC1E 6BT, UK;
Département de Physique Théorique
Université de Genève
1211 Genève 4, Switzerland
d.piras@ucl.ac.uk

Hiranya V. Peiris
Department of Physics & Astronomy
University College London
Gower Street, London WC1E 6BT, UK;
The Oskar Klein Centre for Cosmoparticle Physics
Department of Physics, Stockholm University
AlbaNova, Stockholm, SE-106 91, Sweden
h.peiris@ucl.ac.uk

Andrew Pontzen
Department of Physics & Astronomy
University College London
Gower Street, London WC1E 6BT, UK
a.pontzen@ucl.ac.uk

Luisa Lucie-Smith
Max-Planck-Institut für Astrophysik
Karl-Schwarzschild-Str. 1
85748 Garching, Germany
luisals@mpa-garching.mpg.de

Ningyuan Guo
Department of Physics & Astronomy
University College London
Gower Street, London WC1E 6BT, UK
ningyuan.guo.20@ucl.ac.uk

Brian Nord
Fermi National Accelerator Laboratory;
Kavli Institute for Cosmological Physics &
Department of Astronomy and Astrophysics,
University of Chicago
nord@fnal.gov

Abstract

We develop the use of mutual information (MI), a well-established metric in information theory, to interpret the inner workings of deep learning models. To accurately estimate MI from a finite number of samples, we present GMM-MI, an algorithm based on Gaussian mixture models that can be applied to both discrete and continuous settings. GMM-MI is computationally efficient, robust to hyperparameter choices and provides the uncertainty on the MI estimate due to the finite sample size. We demonstrate the use of our MI estimator in the context of representation learning, working with synthetic data and physical datasets describing highly non-linear processes. We use GMM-MI to quantify both the level of disentanglement between the latent variables, and their association with relevant physical quantities, thus unlocking the interpretability of the latent representation. We make GMM-MI publicly available in this GitHub repository.

1 Introduction

Despite recent progress, deep neural networks remain opaque models, and their power as universal approximators [7, 15, 16] comes at the expense of interpretability [26]. Many techniques have been developed to gain insight into such black-box models [4, 24, 30, 32–34, 36, 37]; however, a general framework to interpret deep neural networks is still an avenue of active investigation [21, 22]. In

*Corresponding author; dpiras.github.io.

Machine Learning and the Physical Sciences workshop, NeurIPS 2022.

this work, we focus on representation learning, where a high-dimensional dataset is compressed to a smaller set of latent variables, and link the latent variables to relevant physical quantities by estimating their mutual information (MI). MI is a well-established information-theoretic measure of the relationship between two random variables which allows us to interpret what the model has learned about the domain-specific parameters: by interrogating the model through MI, we aim to discover what information is used by the model in making predictions, thus achieving the interpretation of its inner workings. We also use MI to quantify the level of disentanglement of the latent variables.

The use of MI estimates for interpreting deep representation learning has recently been investigated [5, 6, 23, 31]. However, estimating the mutual information $I(X, Y)$ between two random variables X and Y, given samples from their joint distribution $p_{(X,Y)}$, remains a long-standing challenge, since it requires $p_{(X,Y)}$ to be known or estimated accurately [27, 35]. Moreover, exploiting MI to interpret deep representation learning requires the uncertainties on the MI estimate to be quantified, ensuring that any trends in MI are statistically significant. To address these requirements, we present GMM-MI (pronounced "Jimmie"), an algorithm to estimate the full distribution of $I(X, Y)$ based on fitting the samples with Gaussian mixture models (GMMs). GMMs represent a flexible, efficient and well-established model to perform density estimation of the samples. We have verified that the error estimates returned by GMM-MI are statistically correct on test datasets including bivariate distributions of various shapes and synthetic GMM datasets. After validating GMM-MI on these toy data, we train representation learning models on high-dimensional datasets, including simulations of dark matter halos, real astrophysical spectra and synthetic shape images with known labels, and demonstrate the use of GMM-MI to achieve the interpretability of such models.

2 Method

2.1 Estimation procedure (GMM-MI)

When X and Y are continuous variables with values over $\mathcal{X} \times \mathcal{Y}$, $I(X, Y)$ is defined as:

$$I(X,Y) \equiv \int_{\mathcal{X}\times\mathcal{Y}} p_{(X,Y)}(x,y) \ln \frac{p_{(X,Y)}(x,y)}{p_X(x)p_Y(y)}\, dx\, dy \,, \tag{1}$$

where p_X and p_Y are the marginal distributions of X and Y, respectively, and $\ln$ refers to the natural logarithm, so that MI is measured in natural units (nat). $I(X, Y)$ represents the amount of information one gains about Y by observing X (or vice versa); a comprehensive summary of MI and its properties can be found in Vergara and Estévez [35]. Our algorithm uses a GMM with c components to obtain a fit of the joint distribution $p_{(X,Y)}$:

$$p_{(X,Y)}(x, y|\theta) = \sum_{i=1}^{c} w_i \mathcal{N}(x, y|\mu_i, \Sigma_i) \,, \tag{2}$$

where θ is the set of weights $w_{1:c}$, means $\mu_{1:c}$ and covariance matrices $\Sigma_{1:c}$. With this choice, the marginals $p(x)$ and $p(y)$ are also GMMs, with parameters determined by θ. The procedure for estimating MI and its associated uncertainty is as follows.

1. For a given number of GMM components c, we initialize n_{init} different GMM models. We obtain each set of initial GMM parameters by randomly assigning the responsibilities, i.e. the probabilities that each point belongs to a component i, sampling from a uniform distribution. The starting μ_i and Σ_i are calculated as the mean and covariance matrix of all points, weighted by the responsibilities; each w_i is initialized as the average responsibility across all points. Other initialization procedures are also implemented in GMM-MI and could alternatively be used.
2. We fit the data using k-fold cross-validation, i.e. we train a GMM on $k - 1$ subsets of the data (or "folds"), and evaluate the trained model on the remaining validation fold. Each fit is performed with the expectation-maximization algorithm [10], and terminates when the change in log-likelihood on the training data is smaller than a chosen threshold. We also add a small regularization constant ω to the diagonal of each covariance matrix to avoid singular covariance matrices [25].
3. We select the model with the highest mean validation log-likelihood across folds $\hat{\ell}_c$, since it has the best generalization performance. Among the k models corresponding to $\hat{\ell}_c$, we also

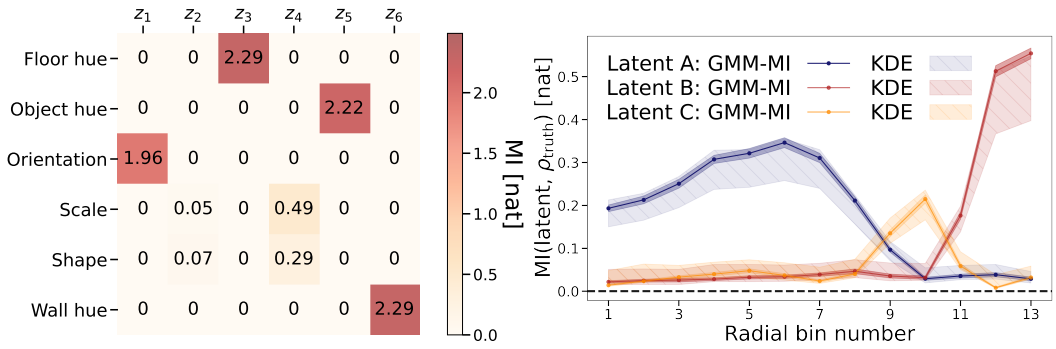


Figure 1: *Left panel*: Mutual information (MI) between each ground truth factor and each latent variable of the β-VAE model trained on the 3D Shapes dataset, obtained using GMM-MI. All zeros indicate values of MI below 0.01 nat. *Right panel*: MI between each latent variable and dark matter halo density ρ_{truth} in each radial bin for the $\mathrm{IVE}_{\mathrm{infall}}$ model [23]. The points with darker error bars correspond to the mean and standard deviation obtained with GMM-MI. The striped areas indicate the kernel density estimate (KDE) values with bandwidths of 0.3 (lower limit) and 0.1 (upper limit).

store the final GMM parameters with the highest validation log-likelihood on a single fold: these will be used to initialize each bootstrap fit in step 5, thus reducing the risk of stopping at local optima and significantly accelerating convergence.

4. We repeat steps 1–3 iteratively increasing the number of GMM components from $c = 1$. We stop when $\hat{\ell}_{c+1} - \hat{\ell}_c$ is smaller than a user-specified positive threshold, and select this value of c as the optimal number of GMM components to fit.
5. We bootstrap the data n_{b} times, and fit a GMM to each bootstrapped realization. Each fit is initialized with the GMM parameters selected in step 3, and with c found in step 4.
6. For each fitted model, we calculate MI by solving the integral in Eq. (1) using Monte Carlo (MC) integration over M samples. We finally return the sample mean and standard deviation of the distribution of MI values.

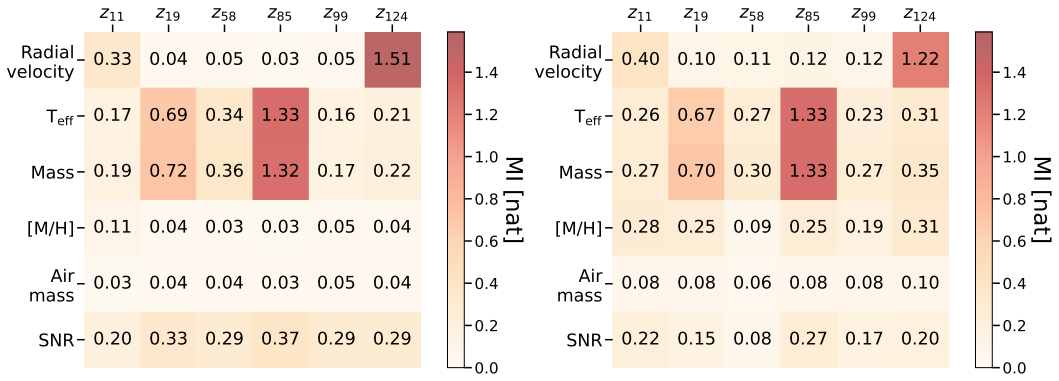
In many instances, the factors of variation that are used to generate the data are discrete variables; in these cases, we will need to estimate MI between a continuous variable X and a categorical variable F which can take v different values $f_{1:v}$. Assuming the v values have equal probability (as for the 3D shapes dataset in Sect. 3.1), the mutual information $I(X, F)$ can be expressed as (the full derivation is in Appendix A):

$$I(X,F) = \frac{1}{v}\sum_{i=1}^{v}\int_{\mathcal{X}} dx\, p_{(X|F)}(x|f_i)\left[\ln p_{(X|F)}(x|f_i) - \ln\frac{1}{v}\sum_{j=1}^{v} p_{(X|F)}(x|f_j)\right], \qquad (3)$$

where we use a GMM to fit each conditional probability $p_{(X|F)}(x|f_i)$. All implementation details are reported in the PYTHON code which can be found in the GMM-MI GitHub repository.

3 Results

We first validate our procedure on toy data for which the ground truth MI is known; the results are shown in Appendix B. We then train representation learning models on synthetic and real data, using our MI estimator to quantify the level of disentanglement of latent variables, and link them to relevant physical parameters. In the following experiments, we consider $k = 3$ folds, $n_{\mathrm{init}} = 5$ initializations, a log-likelihood threshold on each fit of 10^{-5}, $n_{\mathrm{b}} = 100$ bootstrap realizations, $M = 10^5$ MC samples, and a regularization scale of $\omega = 10^{-15}$; we found GMM-MI to be robust to the hyperparameter choice. We run all the experiments on a single CPU node with 40 2.40GHz Intel Xeon Gold 6148 cores using no more than 300 MB of RAM, typically obtaining the full distribution of MI in $\mathcal{O}(10)$ s; since GMMs have linear time complexity in the number of fitted samples and in the number of components [3], GMM-MI results in an efficient algorithm. As representation learning models, we consider β-variational autoencoders (β-VAEs, [14, 18]), where one neural network is



(a) Our results, using GMM-MI. (b) Results from Sedaghat et al. [31], using histograms.

Figure 2: *Left panel*: Mutual information (MI) between the six most-informative latent variables and six astrophysical parameters described in Sect. 3.3, calculated using our algorithm GMM-MI. *Right panel*: Same as the left panel, using the MI estimator of Sedaghat et al. [31] based on histograms.

trained to encode high-dimensional data into a distribution over disentangled latent variables $\mathbf{z}$ (with disentanglement level controlled by β), and a second network decodes samples from the latent distribution back into data points. In Sect. 3.2, we consider the interpretable variational encoder (IVE) [23], where latent samples are combined with the halo radius r through the decoder to predict dark matter halo density profiles at each given r.

3.1 3D Shapes

We consider the 3D Shapes dataset [2, 17], which consists of images of various shapes that were generated by varying six factors including shape, scale and orientation. We train a β-VAE using a 6-dimensional latent space and setting the value of β using cross-validation. After training, we encode the test set (10% of the data, i.e. 48 000 points) and sample one point from each latent distribution. To interpret what each latent variable z_i has learned about each generative factor of variation f_j, we measure the mutual information $I(z_i, f_j)$ using Eq. (3). In the left panel of Fig. 1 we report the MI values for all latents and factors using GMM-MI: except for scale and shape, each latent variable carries information about a single factor of variation. The difficulty in disentangling scale and shape was also reported in Kim and Mnih [17]. To assess the level of dependence between latent variables, we calculate $I(z_i, z_j)$: these values are below 10^{-4} nat for all pairs, except for the one carrying information about both scale and shape, i.e. $I(z_2, z_4) = 0.04 \pm 0.01$ nat.

3.2 Dark matter halo density profiles

Following Lucie-Smith et al. [23, LS22 hereafter], we consider 4332 dark matter halos coming from a single N-body simulation, and encode them using their $\text{IVE}_{\text{infall}}$ model with 3 latent variables. The latent representation is used to predict the dark matter halo density profile in 13 different radial bins. We calculate the MI between the ground-truth halo density in each radial bin and each latent variable. We show the trend of MI for all radial bins and latent variables in the right panel of Fig. 1. We compare the estimates from GMM-MI with those obtained using kernel density estimation (KDE) with different bandwidths, as done in LS22. A major difference between the two approaches is that our bands indicate the error coming from the limited sample size, while their bands represent the sensitivity of the KDE to different bandwidths. The results are in good agreement: in particular, GMM-MI returns estimates closer to the KDE approach with smaller bandwidth when MI is high; in this case, the higher KDE bandwidth value underfits the data. On the other hand, for lower values of MI, GMM-MI yields estimates consistent with the KDE ones at higher bandwidth, since the lower bandwidth overfits the data. We also checked that the latent variables of the $\text{IVE}_{\text{infall}}$ model are independent: as in LS22, the MI between each pair of latents is $\mathcal{O}(10^{-2})$ nat.

3.3 Stellar spectra

We consider the model presented in Sedaghat et al. [31, S21 hereafter], where a β-VAE is trained on about 7000 real unique stellar spectra with a 128-dimensional latent space. Our analysis is carried out on the six most informative latents, selected according to the median absolute deviation (MAD). We calculate MI between the six latents and six known continuous physical factors: the star radial velocity, its effective temperature $T_{\rm eff}$, its mass, its metallicity $[{\rm M}/{\rm H}]$, the atmospheric air mass and the signal-to-noise ratio (SNR). The MI estimates obtained with GMM-MI are shown in the left panel of Fig. 2: the $124^{\rm th}$ latent variable shows high dependence on the radial velocity, while the $85^{\rm th}$ latent appears entangled with both $T_{\rm eff}$ and mass; the other physical parameters do not show a dependence on a particular latent (further discussion can be found in S21). The right panel of Fig. 2 shows the results with the procedure outlined in S21, which uses histograms with a certain number of bins (40 in this case) as density estimators. The trend agrees with our results, even though the particularly high number of bins chosen might overfit the data and overestimate MI (compare e.g. the $[{\rm M}/{\rm H}]$ MI estimates), analogously to the KDE results in Fig. 1. On the other hand, our algorithm provides a robust way to select the hyperparameters, thus avoiding underfitting or overfitting the samples.

4 Conclusions

We presented GMM-MI (pronounced "Jimmie"), an efficient and robust algorithm based on Gaussian mixture models to estimate the distribution of mutual information (MI) between two random variables given samples from their joint distribution. We demonstrated the application of GMM-MI to interpret the latent space of three different deep representation learning models trained on synthetic and real datasets. We calculated both the MI between latent variables and physical factors, and the MI between the latent variables themselves, to quantify their disentanglement. We plan to extend our work by improving the density estimation with more expressive tools such as normalizing flows (NFs, [11, 29]), which can be seamlessly integrated into neural network-based settings and can benefit from graphics processing unit (GPU) acceleration. Moreover, combining NFs with a differentiable numerical integrator would make our estimator amenable to backpropagation, thus allowing its use in the context of MI optimization. GMM-MI is publicly available in this GitHub repository (https://github.com/dpiras/GMM-MI, also accessible by clicking the icon).

Broader impact statement

The application of deep learning (DL) models to a variety of scientific fields and beyond is quickly gaining popularity, and their significant impact on these areas is unquestionable. However, understanding *what* these models are learning and *why* they return their predictions remain open questions, with many routes being currently investigated. Our work proposes the use of the established metric of mutual information to achieve the interpretability and explainability of such models, thus increasing the trust that can be placed in such models. Additionally, our work aims to obtain new physical insights from machine learning, showing that it is possible to gain scientific knowledge from deep learning models, which can then impact the design of new theories and hypotheses.

Acknowledgments and Disclosure of Funding

We thank Nima Sedaghat, Martino Romaniello and Vojtech Cvrcek for sharing the stellar spectra model and data. We are also grateful to Justin Alsing for useful discussions about initialization procedures for GMM fitting. DP was supported by the UCL Provost's Strategic Development Fund, and by a Swiss National Science Foundation (SNSF) Professorship grant (No. 202671). The work of HVP was supported by the Göran Gustafsson Foundation for Research in Natural Sciences and Medicine and the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement no. 101018897 CosmicExplorer). HVP and LLS acknowledge the hospitality of the Aspen Center for Physics, which is supported by National Science Foundation grant PHY-1607611. The participation of HVP and LLS at the Aspen Center for Physics was supported by the Simons Foundation. This study was supported by the European Union's Horizon 2020 research and innovation programme under grant agreement No. 818085 GMGalaxies. AP is additionally supported by the Royal Society. NG was funded by the UCL Graduate Research

Scholarship (GRS) and UCL Overseas Research Scholarship (ORS). This manuscript has been authored by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics. This work used computing equipment funded by the Research Capital Investment Fund (RCIF) provided by UKRI, and partially funded by the UCL Cosmoparticle Initiative. This work used facilities provided by the UCL Cosmoparticle Initiative.

References

[1] M. I. Belghazi, A. Baratin, S. Rajeshwar, S. Ozair, Y. Bengio, A. Courville, and D. Hjelm. Mutual Information Neural Estimation. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 531–540. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/belghazi18a.html.

[2] C. Burgess and H. Kim. 3D Shapes Dataset. https://github.com/deepmind/3d-shapes, 2018.

[3] V. Chandola, R. R. Vatsavai, D. Kumar, and A. Ganguly. Chapter 10 - analyzing big spatial and big spatiotemporal data: A case study of methods and applications. In V. Govindaraju, V. V. Raghavan, and C. Rao, editors, *Big Data Analytics*, volume 33 of *Handbook of Statistics*, pages 239–258. Elsevier, 2015. doi: https://doi.org/10.1016/B978-0-444-63492-4.00010-1. URL https://www.sciencedirect.com/science/article/pii/B9780444634924000101.

[4] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian. Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847, 2018. doi: 10.1109/WACV.2018.00097.

[5] R. T. Q. Chen, X. Li, R. B. Grosse, and D. K. Duvenaud. Isolating Sources of Disentanglement in Variational Autoencoders. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/1ee3dfcd8a0645a25a35977997223d22-Paper.pdf.

[6] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. *Advances in Neural Information Processing Systems*, 29, 2016.

[7] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems (MCSS)*, 2(4):303–314, Dec. 1989. ISSN 0932-4194. doi: 10.1007/BF02551274. URL http://dx.doi.org/10.1007/BF02551274.

[8] G. Darbellay and I. Vajda. Estimation of the information by an adaptive partitioning of the observation space. *IEEE Transactions on Information Theory*, 45(4):1315–1321, 1999. doi: 10.1109/18.761290.

[9] G. Darbellay and I. Vajda. Entropy expressions for multivariate continuous distributions. *IEEE Transactions on Information Theory*, 46(2):709–712, 2000. doi: 10.1109/18.825848.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. ISSN 00359246. URL http://www.jstor.org/stable/2984875.

[11] L. Dinh, D. Krueger, and Y. Bengio. NICE: Non-linear Independent Components Estimation. *arXiv e-prints*, art. arXiv:1410.8516, Oct. 2014.

[12] M. Donsker and S. Varadhan. Asymptotic evaluation of certain Markov process expectations for large time. IV. *Communications on Pure and Applied Mathematics*, 36(2):183–212, Mar. 1983. ISSN 0010-3640. doi: 10.1002/cpa.3160360204.

[13] M. A. Haeri and M. M. Ebadzadeh. Estimation of mutual information by the fuzzy histogram. *Fuzzy Optim. Decis. Mak.*, 13(3):287–318, 2014. doi: 10.1007/s10700-014-9178-0. URL https://doi.org/10.1007/s10700-014-9178-0.

[14] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *ICLR*, 2017.

[15] K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2): 251–257, 1991. ISSN 0893-6080. doi: https://doi.org/10.1016/0893-6080(91)90009-T. URL https://www.sciencedirect.com/science/article/pii/089360809190009T.

[16] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989. ISSN 0893-6080. doi: https://doi.org/10.1016/0893-6080(89)90020-8. URL https://www.sciencedirect.com/science/article/pii/0893608089900208.

[17] H. Kim and A. Mnih. Disentangling by Factorising. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2649–2658. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/kim18b.html.

[18] D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[19] L. F. Kozachenko and N. N. Leonenko. Sample estimate of the entropy of a random vector. *Probl. Inf. Transm.*, 23(1-2):95–101, 1987. ISSN 0032-9460.

[20] A. Kraskov, H. Stögbauer, and P. Grassberger. Estimating mutual information. *Phys. Rev. E*, 69:066138, Jun 2004. doi: 10.1103/PhysRevE.69.066138. URL https://link.aps.org/doi/10.1103/PhysRevE.69.066138.

[21] X. Li, H. Xiong, X. Li, X. Wu, X. Zhang, J. Liu, J. Bian, and D. Dou. Interpretable Deep Learning: Interpretation, Interpretability, Trustworthiness, and Beyond. *arXiv e-prints*, art. arXiv:2103.10689, Mar. 2021.

[22] P. Linardatos, V. Papastefanopoulos, and S. Kotsiantis. Explainable AI: A Review of Machine Learning Interpretability Methods. *Entropy*, 23(1), 2021. ISSN 1099-4300. doi: 10.3390/e23010018. URL https://www.mdpi.com/1099-4300/23/1/18.

[23] L. Lucie-Smith, H. V. Peiris, A. Pontzen, B. Nord, J. Thiyagalingam, and D. Piras. Discovering the building blocks of dark matter halo density profiles with neural networks. *Phys. Rev. D*, 105:103533, May 2022. doi: 10.1103/PhysRevD.105.103533. URL https://link.aps.org/doi/10.1103/PhysRevD.105.103533.

[24] S. M. Lundberg and S.-I. Lee. A Unified Approach to Interpreting Model Predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[25] P. Melchior and A. D. Goulding. Filling the gaps: Gaussian mixture models from noisy, truncated or incomplete samples. *Astronomy and Computing*, 25:183–194, Oct. 2018. doi: 10.1016/j.ascom.2018.09.013.

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

[27] L. Paninski. Estimation of Entropy and Mutual Information. *Neural Computation*, 15(6):1191–1253, 06 2003. ISSN 0899-7667. doi: 10.1162/089976603321780272. URL https://doi.org/10.1162/089976603321780272.

[28] B. Poole, S. Ozair, A. van den Oord, A. A. Alemi, and G. Tucker. On Variational Bounds of Mutual Information. *arXiv e-prints*, art. arXiv:1905.06922, May 2019.

[29] D. J. Rezende and S. Mohamed. Variational Inference with Normalizing Flows. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 1530–1538. JMLR.org, 2015.

[30] M. T. Ribeiro, S. Singh, and C. 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*, KDD '16, page 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939778. URL https://doi.org/10.1145/2939672.2939778.

[31] N. Sedaghat, M. Romaniello, J. E. Carrick, and F.-X. Pineau. Machines learn to infer stellar parameters just by looking at a large number of spectra. *Monthly Notices of the Royal Astronomical Society*, 501(4):6026–6041, Mar. 2021. doi: 10.1093/mnras/staa3540.

[32] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017. doi: 10.1109/ICCV.2017.74.

[33] A. Shrikumar, P. Greenside, and A. Kundaje. Learning Important Features through Propagating Activation Differences. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3145–3153. JMLR.org, 2017.

[34] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *In Workshop at International Conference on Learning Representations*, 2014.

[35] J. R. Vergara and P. A. Estévez. A Review of Feature Selection Methods Based on Mutual Information. *arXiv e-prints*, art. arXiv:1509.07577, Sept. 2015.

[36] M. D. Zeiler and R. Fergus. Visualizing and Understanding Convolutional Networks. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 818–833, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10590-1.

[37] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning Deep Features for Discriminative Localization. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2929, Los Alamitos, CA, USA, jun 2016. IEEE Computer Society. doi: 10.1109/CVPR.2016.319. URL https://doi.ieeecomputersociety.org/10.1109/CVPR.2016.319.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] Sect. 3 and Appendix B include the main results as indicated in the abstract and introduction.
   (b) Did you describe the limitations of your work? [Yes] See Sect. 4, where we outline the current limitations and how we plan to address them.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] Eq. 3 is not a novel result, but we included its derivation in Appendix A.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We shared a link to the GitHub repository.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No] We provide https links where the license is specified.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] GMM-MI is available in this GitHub repository.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] The data we are using are either publicly available, or produced by us.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Derivation of the mutual information between a continuous and a categorical variable

In order to derive Eq. (3), we first rewrite Eq. (1) as:

$$I(X,Y) = \int_{\mathcal{X}\times\mathcal{Y}} p_{(X|Y)}(x|y)p_Y(y) \ln \frac{p_{(X|Y)}(x|y)}{p_X(x)}\, dx\, dy\,. \tag{4}$$

Then, we assume a generalized probability density function for the categorical variable F over $\mathcal{F}$:

$$p_F(f) = \sum_{i=1}^{v} p_F(f=f_i)\delta(f-f_i) = \frac{1}{v}\sum_{i=1}^{v}\delta(f-f_i)\,, \tag{5}$$

where δ is the Dirac delta function, and in the last step we assumed that F can take the values $f_{1:v}$ with equal probability. Combining the last two equations, we obtain:

$$\begin{aligned}
I(X,F) &= \int_{\mathcal{X}\times\mathcal{F}} dx\, df\; p_{(X|F)}(x|f)p_F(f) \ln \frac{p_{(X|F)}(x|f)}{p_X(x)} \\
&= \frac{1}{v}\sum_{i=1}^{v}\int_{\mathcal{X}\times\mathcal{F}} dx\, df\; p_{(X|F)}(x|f)\delta(f-f_i)\left[\ln p_{(X|F)}(x|f) - \ln p_X(x)\right] \\
&= \frac{1}{v}\sum_{i=1}^{v}\int_{\mathcal{X}} dx\; p_{(X|F)}(x|f_i)\left[\ln p_{(X|F)}(x|f_i) - \ln p_X(x)\right] \\
&= \frac{1}{v}\sum_{i=1}^{v}\int_{\mathcal{X}} dx\; p_{(X|F)}(x|f_i)\left[\ln p_{(X|F)}(x|f_i) - \ln \frac{1}{v}\sum_{j=1}^{v} p_{(X|F)}(x|f_j)\right]\,,
\end{aligned} \tag{6}$$

as reported in Eq. (3).

B Validation of GMM-MI

In order to validate our algorithm, we compare it with two established estimators of MI. The KSG estimator, first proposed in Kraskov et al. [20], rewrites MI as a function of the Shannon entropy, and uses the Kozachenko-Leonenko estimator [19] to evaluate it. In our experiments, we consider the implementation of the KSG estimator available from SKLEARN at this https link. We also compare our algorithm against the MINE estimator proposed in Belghazi et al. [1], based on neural networks. MINE interprets MI as the KL divergence between the joint distribution and the product of the marginals, and then considers its Donsker-Varadhan representation [12]. In our experiments, we consider the implementation available at this https link. Further tests validating GMM-MI on synthetic GMM datasets are not presented here for conciseness.

We consider three bivariate distributions: a Gaussian distribution with unit variance of each marginal and varying level of correlation $\rho \in [-1,1]$, the gamma-exponential distribution [8, 9, 13, 20] and the ordered Weinman exponential distribution [8, 9, 13, 20]. The true values of MI for these distributions can be obtained via direct integration of Eq. (1). In the bivariate Gaussian case:

$$I(X,Y)_{\text{true}} = -\frac{1}{2}\ln\left(1-\rho^2\right)\,, \tag{7}$$

where ρ is the correlation coefficient. For the gamma-exponential distribution, which has density:

$$p_{(X,Y)}(x,y|\alpha) = \begin{cases} \frac{1}{\Gamma(\alpha)}x^{\alpha}e^{-x-xy} & x>0, y>0 \\ 0 & \text{otherwise} \end{cases}\,, \tag{8}$$

where $\alpha > 0$ is a free parameter and Γ is the gamma function, MI is calculated as:

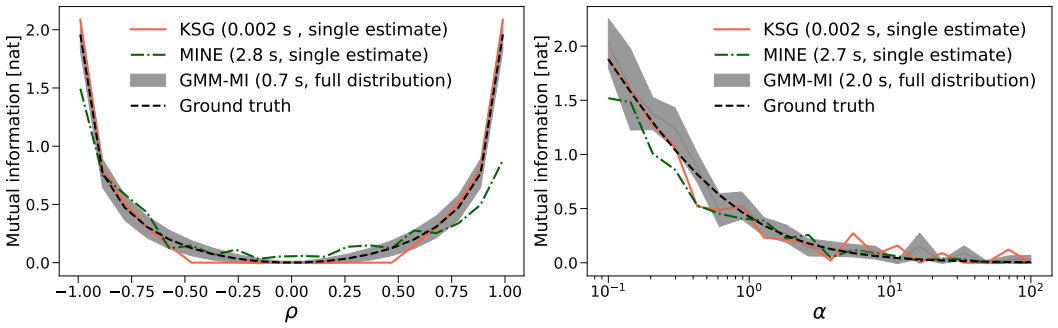
$$I(X,Y) = \psi(\alpha+1) - \ln\alpha\,, \tag{9}$$

where ψ is the digamma function, defined as:

$$\psi(x) = \frac{\Gamma'(x)}{\Gamma(x)}\,. \tag{10}$$

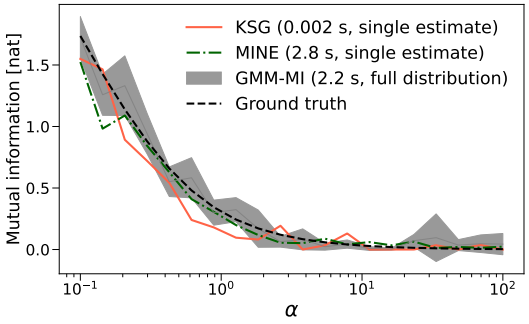
For the ordered Weinman exponential distribution, with density:

$$p_{(X,Y)}(x,y|\alpha) = \begin{cases} \frac{2}{\alpha}e^{-2x-\frac{y-x}{\alpha}} & y>x>0 \\ 0 & \text{otherwise} \end{cases}\,, \tag{11}$$



(a) Bivariate Gaussian distribution.

(b) Gamma-exponential distribution.



(c) Ordered Weinman exponential distribution.

Figure 3: Value of mutual information (MI) for: (a) a bivariate Gaussian distribution with varying correlation coefficient ρ; (b) a gamma-exponential distribution with varying α, as in Eq. (8); (c) an ordered Weinman exponential distribution with varying α, as in Eq. (11). The dashed black line indicates the ground truth MI. We compare the KSG estimator ([20], solid red line), the MINE estimator ([1], dotted-dash green line), and our estimator GMM-MI, indicated with the gray shaded area (mean $\pm$ two standard deviations). The numbers in parentheses indicate the time to obtain a single estimate with KSG and MINE, and the full distribution of MI in the case of GMM-MI, for each ρ.

where $\alpha > 0$ is a free parameter, MI reads:

$$I(X,Y) = \begin{cases} \ln\left(\frac{1-2\alpha}{2\alpha}\right) + \psi\left(\frac{1}{1-2\alpha}\right) - \psi(1) & \alpha < \frac{1}{2} \\ -\psi(1) & \alpha = \frac{1}{2} \\ \ln\left(\frac{2\alpha-1}{2\alpha}\right) + \psi\left(\frac{2\alpha}{2\alpha-1}\right) - \psi(1) & \alpha > \frac{1}{2} \end{cases} . \tag{12}$$

Since $I(X,Y)$ is invariant under invertible transformations of each random variable, we consider $\ln(X)$ and $\ln(Y)$ when estimating MI in the case of the last two distributions [20]. To demonstrate the power of our estimator, we restrict ourselves to the case with only $N = 200$ samples. To estimate MI, we consider the KSG estimator with 1 neighbor (to minimize the bias), the MINE estimator trained for 50 epochs with a learning rate of 10^{-3} and a batch size of 32, and our estimator GMM-MI with $k = 2$ folds, $n_{\text{init}} = 3$ different initializations, a log-likelihood threshold on each individual fit of 10^{-5}, a threshold on the mean validation log-likelihood to select the number of GMM components of 10^{-5}, $n_{\text{b}} = 100$ bootstrap realizations, $M = 10^4$ MC samples, and a regularization scale of $\omega = 10^{-12}$.

The results are reported in Fig. 3. The KSG estimator is the fastest, and yields MI values closely matching the ground truth, but returns biased estimates around e.g. $|\rho| = 0.4$ in the bivariate Gaussian case, and $\alpha = 1$ in the ordered Weinman case. The MINE estimator is more computationally expensive and shows a relatively high variance, which is expected since MINE has been shown to be prone to variance overestimation due to the use of batches [28]. GMM-MI, on the other hand, returns a distribution of MI in good agreement with the ground truth in $\mathcal{O}(1)$ s, and includes an uncertainty estimate due to the finite sample size. We also found the results of GMM-MI to be robust to the choice of hyperparameters: changing the values of the likelihood threshold, MC samples, bootstrap realizations or regularization scale by one order of magnitude, or doubling the number of folds and initializations, did not significantly change the results obtained with GMM-MI.