Feature Attribution for Deep Learning Models through Total Variance Decomposition

Yinzhu Jin[1] YJ3CZ@VIRGINIA.EDU
[1] *Dept. of Computer Science, University of Virginia, Charlottesville, VA, USA*
Shen Zhu[1,2] SZ9JT@VIRGINIA.EDU
[2] *Dept. of Electrical & Computer Engineering, University of Virginia, Charlottesville, VA, USA*
P. Thomas Fletcher[1,2] PTF8V@VIRGINIA.EDU

Editors: Accepted for publication at MIDL 2025

Abstract

This paper introduces a new approach to feature attribution for deep learning models, quantifying the importance of specific features in model decisions. By decomposing the total variance of model decisions into explained and unexplained fractions, conditioned on the target feature, we define the feature attribution score as the proportion of explained variance. This method offers a solid statistical foundation and normalized quantitative results. When ample data is available, we compute the score directly from test data. For scarce data, we use constrained sampling with generative diffusion models to represent the conditional distribution at a given feature value. We demonstrate the method's effectiveness on both a synthetic image dataset with known ground truth and OASIS-3 brain MRIs.

Keywords: Feature attribution, counterfactual explanation, generative diffusion model.

1. Introduction

Deep learning has achieved remarkable performance in image classification by leveraging complex neural network architectures to automatically extract and learn features. However, despite its success, deep learning often operates as a "black box," where the internal workings and decision-making processes of the models are challenging to interpret. Numerous methods have been proposed to interpret model decisions, with saliency maps (Selvaraju et al., 2017; Sundararajan et al., 2017; Schulz et al., 202) and counterfactual explanations (CE) (Wachter et al., 2017; Goyal et al., 2019b; Augustin et al., 2022) emerging as two of the most widely used techniques. Both strategies are great for discovering important features without any background knowledge. Nevertheless, for medical imaging tasks involving significant human expertise, explanations relying on understandable feature contributions are preferred. In contrast, techniques like CE that are designed to mimic human reasoning, do not appear to enhance trust in the system's predictions (Wang and Yin, 2021).

We propose a metric to quantify a classifier's reliance on a specific target feature by decomposing the variance of model predictions. Our score reflects the proportion of prediction variance explained by the feature, based on its conditional distribution. Unlike saliency maps, which assign scores to image positions, our method applies to any feature with a learnable distribution. For example, in classification of Alzheimer's disease from brain MRI, our model is able to evaluate the importance of both the location of hippocampal voxels and the overall hippocampal volume. While large datasets often provide direct

sampling from the data distribution conditioned on discrete features, sampling conditioned on a continuous feature is not directly possible. To address this, we use diffusion models (Ho et al., 2020) and guided sampling (Chung et al., 2023) to model the conditional distribution.

In summary, our proposed evaluation metrics offer the following advantages:

- Quantified importance evaluation: provides a measurable assessment of feature importance, which is particularly useful in tasks requiring human expertise.
- Broad applicability: applicable to any learnable or annotated features, both continuous and discrete, without relying on classifier robustness.
- Rooted in causality: based on principles of causality theory by observing outcomes when interfering with specific target features.

2. Background

We first introduce related interpretability techniques, then the diffusion model used for learning and sampling from conditional distributions.

2.1. Causality based interpretation

Counterfactual explanations provide intuitive insights by generating a new sample that flips the model's decision with minimal changes to the original image. Early methods composited features from distractor images (Goyal et al., 2019b), while recent approaches use generative models like diffusion models (Ho et al., 2020) for better image quality. These methods often rely on classifier gradients to minimize distance (Augustin et al., 2022; Jeanneret et al., 2022), but when applied to non-robust classifiers, they may generate adversarial examples. While Augustin et al. (2022) can evaluate non-robust classifiers, they still rely on a robust classifier to mitigate this issue.

Other works interpret classifiers using features beyond pixels. For example, CaCE (Goyal et al., 2019a) examines model predictions by varying feature values. Their metric calculates the difference in model outputs when a binary feature is set to negative versus positive. This approach is inherently limited to binary features. In contrast, Jin et al. (2024) proposed attributing continuous features by neutralizing their influence, which they achieve by adjusting feature values to a baseline. However, the choice of this baseline value is not well justified. Both methods rely on variational autoencoders (VAEs) (Kingma, 2013), thus missing out on the advancements offered by state-of-the-art generative models.

2.2. Diffusion models and guided sampling

In a diffusion model (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021), the forward process is a Markov process where Gaussian noise is added gradually to the original data x_0. At each time step t, x_t is sampled from the distribution:

$$q(x_t \mid x_{t-1}) := \mathcal{N}(\sqrt{1-\beta_t}x_{t-1}, \beta_t \mathbf{I}). \tag{1}$$

The time variance schedule β_t ensures the final distribution is approximately a standard Gaussian: $q(x_T \mid x_0) \approx \mathcal{N}(\mathbf{0}, \mathbf{I})$. In the reverse process, the goal is to learn $p_\theta(x_{t-1} \mid x_t)$, the distribution of x_{t-1} given x_t, parameterized by θ.

In DDPM (Ho et al., 2020), the problem is simplified to predicting the added noise ϵ based on x_t and t, formulated as $\epsilon_\theta(x_t, t)$. Alternatively, this can be viewed as a score based model with the score function:

$$\nabla_{x_t} \log p_\theta(x_t) = -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(x_t, t), \tag{2}$$

where $\bar{\alpha}_t := \prod_i^t \alpha_i$ and $\alpha_i := 1 - \beta_i$ (Dhariwal and Nichol, 2021; Song et al., 2021).

To draw samples from a conditional distribution given condition c, we look into the conditional score function $\nabla_{x_t} \log p_\theta(x_t \mid c) = \nabla_{x_t} \log p_\theta(x_t) + \nabla_{x_t} \log p_\theta(c \mid x_t)$. The second term could be the gradient of the classifier that predicts c (Dhariwal and Nichol, 2021). The diffusion posterior sampling (DPS) (Chung et al., 2023) method generalizes to continuous conditions by replacing $p_\theta(c \mid x_t)$ from a classifier to a hypothetical Gaussian distribution:

$$\nabla_{x_t} \log p_\theta(c = c_0 \mid x_t) = -\rho \nabla_{x_t} \|c_0 - g(\hat{x}_0)\|_2^2, \tag{3}$$

with ρ being a constant coefficient, c_0 being the given condition value, $\hat{x}_0$ being the expected x_0 given x_t, and g being the mapping from the data to the feature.

3. Methods

Our goal is to assess a classifier f which is a mapping from input data $X \sim p(X)$ to $\{0, 1\}$. We denote the classifier prediction as $Y := f(X)$, which can be seen as a random variable. Similarly, we define a feature as the output of a mapping g from the input data to some feature value $V := g(X)$, which is another random variable. In our framework, the feature type can be very general, e.g., V may be discrete, continuous, or multivariate.

3.1. Causal model

We represent the causal relationship of the variables involved in our analysis in Figure 1. The variable V represents the target feature, and Y represents the model prediction as defined above. Additionally, W represents the exogenous variables besides V that are present in the data and may affect the model prediction.

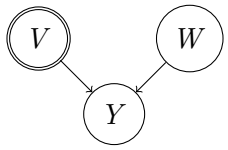


Figure 1: The structural causal model. The double circle represents the variable being set.

Ideally, we would fix W and observe how Y changes with respect to different values of the feature of interest, V. In practice, however, W is difficult to define, not tractable, and hard to control precisely. While one might try to fix W by modifying the data along the direction of ∇g, completely disentangling features remains challenging. We illustrate this with an example using the OASIS-3 dataset from Section 4.1, consisting of 3D brain MRIs cropped around the hippocampus. The feature of interest, hippocampal volume, is estimated using a CNN regression model, g. We applied a DDIM encoder with guided

denoising using the gradient of the trained regressor to generate a series of samples differing only in hippocampal volume. As shown in Figure 2, while the hippocampal volume changes as intended, surrounding structures also change.

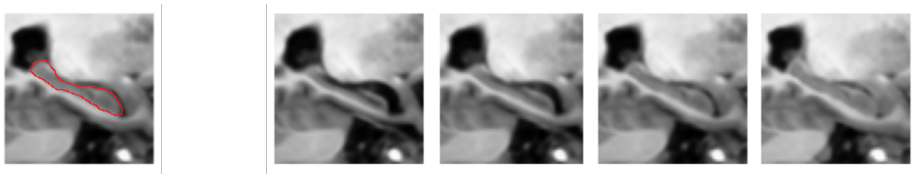


Figure 2: Right: samples of DDIM encoding and guided sampling for enlarging the hippocampus. Left: an illustration of the hippocampus, outlined in red.

On the other hand, we can measure V much more precisely if we have a good estimation of g. In such a case, it is more feasible to fix V up to a measurable amount of error and perturb W randomly. With this strategy, we now design a quantitative score that measures the effect of V on Y.

3.2. Feature importance score

To design our feature importance score, recall a basic theorem in probability theory where the variance is decomposed into two parts using the conditional distribution:

Theorem 1 (Law of total variance (Fox, 2015)) *If A and B are random variables on the same probability space, and the variance of A is finite, then*

$$\mathrm{Var}(A) = \mathrm{E}[\mathrm{Var}(A \mid B)] + \mathrm{Var}[\mathrm{E}(A \mid B)]. \tag{4}$$

The two terms on the right-hand side are often known as the "unexplained" and the "explained" components of the variance, respectively.

Now let's consider decomposing the variance of the classifier prediction Y using the distribution conditioned on the target feature V. We define our feature importance score over the dataset (namely, global score) as the fraction of explained variance:

$$\mathrm{Score}_V := \frac{\mathrm{Var}[\mathrm{E}(Y \mid V)]}{\mathrm{Var}(Y)} = 1 - \frac{\mathrm{E}[\mathrm{Var}(Y \mid V)]}{\mathrm{Var}(Y)}. \tag{5}$$

We can verify that the proposed score always lies within the interval $[0, 1]$ because both terms in Equation 4 are non-negative.

To intuitively understand how this score reflects the importance of a target feature, consider two extreme scenarios. First, if Y is independent of V, then the conditional distribution $P(Y \mid V)$ is the same as $P(Y)$, which means $\mathrm{Var}(Y \mid V) = \mathrm{Var}(Y)$. In this case, the score Score_V will be 0. Conversely, if Y is fully determined by V, then $\mathrm{Var}(Y \mid V) = 0$, resulting in $\mathrm{Score}_V = 1$.

We further extend the feature importance score by introducing a score for each point $V = v_0$ (namely, local score):

$$\mathrm{Score}_{V=v_0} := 1 - \frac{\mathrm{Var}(Y \mid V = v_0)}{\mathrm{Var}(Y)}. \tag{6}$$

This extension is consistent with our global score, as shown in the following equation:

$$\mathrm{E}_{v_0}[\mathrm{Score}_{V=v_0}] = 1 - \frac{\mathrm{E}_{v_0}\left[\mathrm{Var}(Y \mid V = v_0)\right]}{\mathrm{Var}(Y)} = \mathrm{Score}_V. \tag{7}$$

Similarly, the local score indicates the importance of a feature when it takes a specific value. If the feature V is informative about Y, the variance of Y should decrease, leading to a higher local score. This local score is similar to the R^2 score (Pearson, 1901) widely used in regression analysis. Like the R^2 score, it has an upper bound of 1, but unlike the R^2 score, it can also fall below 0. A negative local score suggests that when V takes this certain value, the variability in the classifier's predictions is greater than the variability over the dataset.

3.3. Sampling from conditional distribution

To calculate the proposed score, we need to sample from the conditional distribution $P(Y \mid V = v_0)$. Ideally, we would like to have enough real samples to represent this distribution at every v_0. When V is a categorical feature, and if we are given large enough test set, this can easily be satisfied. However, for continuous V, we might not have enough samples with $V \in (v_0 - \epsilon, v_0 + \epsilon)$ for some small $\epsilon > 0$. In this scenario, we propose to use a generative model to obtain new samples. Although there is no restriction for the type of generative models, we adopt diffusion models (Ho et al., 2020) for their high sample quality. As introduced in Section 2.2, we use a guided sampling method to constrain V.

When performing guided sampling, $\nabla_{x_t} \log p_\theta(x_t)$ is equivalent to performing the usual DDPM denoising step, while Equation 3 is the extra drift term of DPS method (Chung et al., 2023). The feature mapping g can be a regression model trained on the training set given annotations. We also experimented with normalizing the gradient $\nabla_{x_t}\|c_0 - g(\hat{x}_0)\|_2^2$ similar to previous work that applied normalization to the classifier guidance (Augustin et al., 2022). The normalization helps stabilize the sampling.

In conclusion, at each time step t, the guided denoising operation is:

$$\begin{aligned}
\hat{x}_0 &= \frac{1}{\sqrt{\bar{\alpha}_t}}\left(x_t - (1-\bar{\alpha}_t)\epsilon_\theta(x_t, t)\right), \quad z \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \\
x'_{t-1} &= \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\hat{x}_0 + \sigma_t z, \\
x_{t-1} &= x'_{t-1} - \rho \frac{\nabla_{x_t}\|c_0 - g(\hat{x}_0)\|_2^2}{\|\nabla_{x_t}\|c_0 - g(\hat{x}_0)\|_2^2\|_2}.
\end{aligned} \tag{8}$$

The first three steps are regular DDPM denoising with noise standard deviation $\sigma_t = \sqrt{\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t}$. And the last step guides the samples closer to the constraint. Due to the stochasticity of diffusion models, a small portion of samples will still fall far away from the constraint. We remove these samples by examining the resulting V value using g. Note that we perform guided sampling using feature mapping rather than the evaluated classifier, avoiding the risk of generating adversarial samples when using a non-robust classifier.

4. Experiments

We evaluate our proposed metric on one synthetic image dataset and one medical image dataset, and compare it with the diffusion model-based counterfactual explanation

method (Augustin et al., 2022) ("diffusion CE" for short). Both methods use the same diffusion models, but unlike their approach, our experiments use classifiers not specifically trained for robustness. All experiments are implemented with PyTorch (Paszke et al., 2017).

4.1. Datasets

Ellipse dataset is a synthetic dataset generated using the package by Jin (2022). It contains 10,000 images of white ellipses on black backgrounds, varying in position, orientation, size, and aspect ratio. The images are labeled into two categories based on the ellipses' aspect ratio. We use aspect ratio and size, an irrelevant feature, as our target features.
Brain MRI ROI data is from the OASIS-3 dataset (LaMontagne et al., 2019), consisting of 929 subjects diagnosed as cognitively normal (CN) or with Alzheimer's Disease (AD), each with one MRI session. The data was stratified, with 186 subjects for testing. For each subject, we extracted two $64 \times 64 \times 64$ regions of interest (ROIs) centered on the hippocampi, mirroring the right hippocampus along the sagittal plane. The dataset is imbalanced, with 77.5% CN samples. We focus on conditioning with fixed hippocampal voxels or hippocampal volume, as both are known to relate to AD (Sarica et al., 2018; Zhu et al., 2024).

4.2. Results

We now present and discuss the scores obtained using our method. Details on the models, training, and sampling process, together with more samples are provided in the appendix.

4.2.1. Ellipse

Since the ellipse dataset is a synthetic dataset, we know that the only important feature is the aspect ratio. Figure 3 shows samples generated with constraining either the aspect ratio or the size. We computed our scores for aspect ratio and size features. From the global scores reported in Table 1, we can see that our metric can indeed reflect this. Local scores are also reported in Figure 5.

Table 1: Global scores of ellipse classifier.

Feature	Global score
Aspect ratio	0.871
Size	0.049

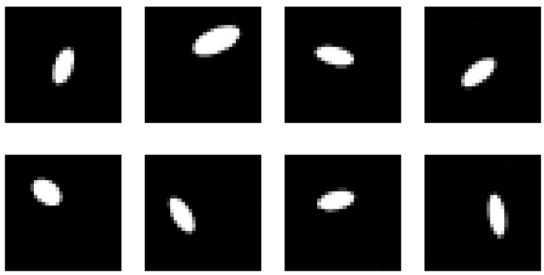


Figure 3: Ellipse samples generated with constrained aspect ratios (top) and sizes (bottom).

The diffusion CE led to 82.5% of samples flipping model predictions. Some original and counterfactual image pairs with flipped predictions are shown in Figure 4. While aspect ratios generally change, size - a feature known to be unimportant—sometimes changes

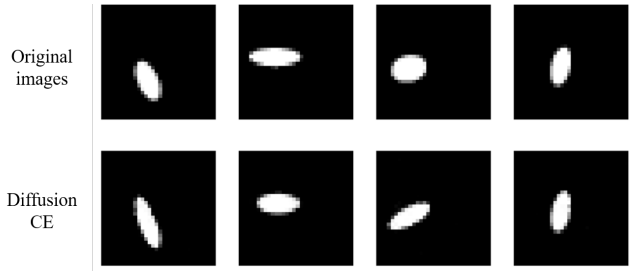


Figure 4: Sample pairs of original ellipse images and their diffusion CE.

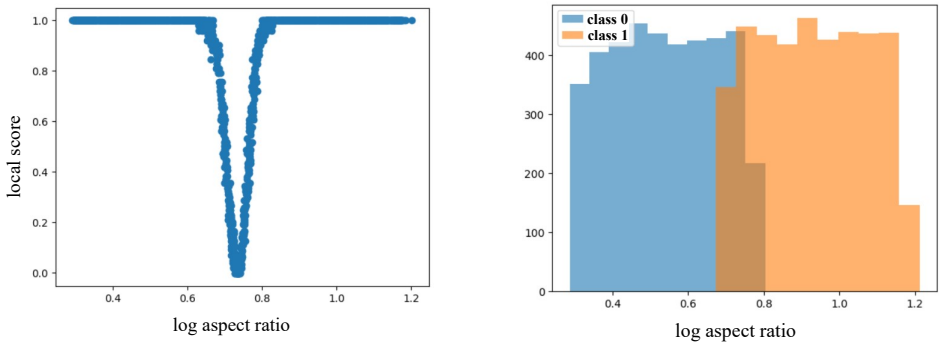


Figure 5: Local scores of the aspect ratio feature in the ellipse data (left) and the distribution of the log aspect ratio values in the training data (right).

as well (see the second column). This mirrors the findings in Figure 2, highlighting the difficulty of changing an image in the gradient direction without affecting other dimensions.

4.2.2. Brain ROI

For the brain ROI dataset, hippocampal volume is constrained using the trained regression model, while the hippocampus is constrained by masking areas outside the hippocampus using the ground truth segmentation, akin to an inpainting task. Random samples with constrained hippocampus are shown in Figure 6, where the hippocampus (in red) closely matches the original, while surrounding brain areas vary. More examples are in Figure 14.

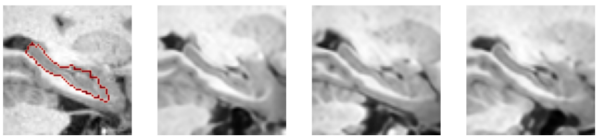


Figure 6: Original image (left most) with hippocampus illustrated in red, and randomly generated samples with constrained hippocampus.

Table 2: Global scores of AD classifier.

Feature	Global score
Hippocampal volume	0.280
Hippocampus	0.448

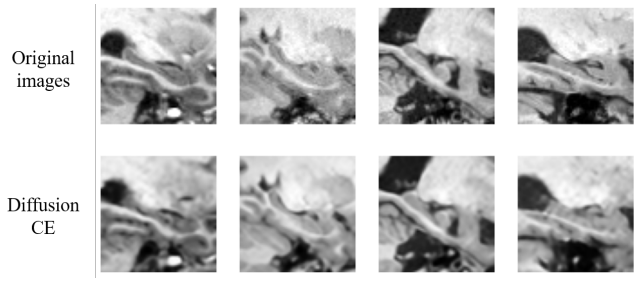


Figure 7: Sample pairs of original brain ROI images and their diffusion CE. First two columns are samples flipping from CN to AD, and the last two from AD to CN.

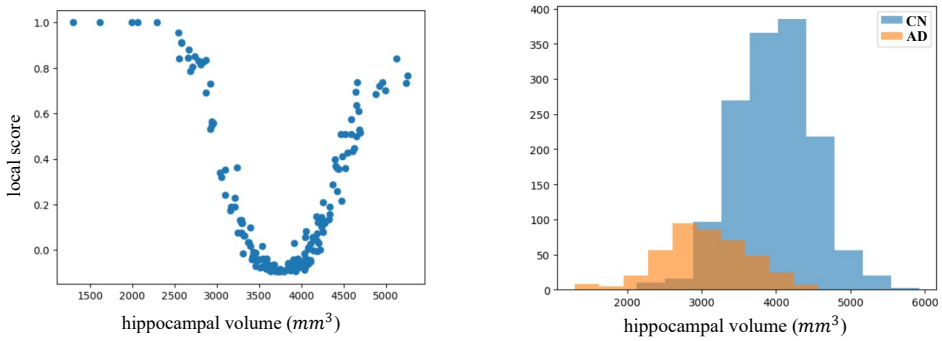


Figure 8: Local scores for the hippocampal volume feature in the brain ROI dataset (left) and the distribution of hippocampal volumes in the training data (right).

Global scores are reported in Table 2, and local scores for corresponding hippocampal volumes are shown on the left side of Figure 8. The global score was computed by averaging the separately calculated expectations for the AD and CN groups, addressing class imbalance, similar to balanced accuracy. These scores suggest that hippocampal volume is a useful classifier feature, with the entire hippocampus explaining more variation in outputs, while the remainder may reflect other factors, like ventricle volume. The local scores are higher when the volume is notably small or large, as expected, since this strongly indicates AD or CN, as shown in the per-class volume distributions on the right side.

Diffusion CE successfully flipped model decisions for all test samples. We attribute this to the VAE-enhanced classifier guidance, as discussed in Appendix C.4. Sample pairs are shown in Figure 7, where changes in hippocampal and ventricle volumes are visible, but subtle. Thus, the CE method aligns with our findings: while CE offers an intuitive explanation, our metric provides a quantitative assessment.

4.3. Sampling evaluation

We assess the effectiveness of constrained sampling methods by evaluating data coverage and estimating the sampling variances of our scores.

Table 3: Coverage score of different sampling methods.

(a) The ellipse data

Sampling method	Coverage		
	k=5	k=3	k=1
Plain DDPM	0.978	0.972	0.887
Constrained aspect ratio	0.991	0.975	0.887
Constrained size	0.991	0.978	0.905

(b) The brain MRI ROI data

Sampling method	Coverage	
	k=3	k=1
Plain DDPM	1.000	0.994
Constrained hippo. vol.	1.000	1.000
Constrained hippo.	1.000	1.000

To evaluate sample coverage of the real data distribution, we combine samples from different conditions and compare them to the ground truth test set. As a baseline, we use plain DDPM sampling. The coverage metric from Naeem et al. (2020) measures the fraction of real samples with generated samples in their k-nearest neighborhood using L_2 distance in the embedding space. We used a VGG16 (Simonyan and Zisserman, 2015) encoder for the ellipse dataset and our own VAE encoder for the brain ROI dataset. Results in Table 3 show our method slightly outperforms plain DDPM sampling, likely due to real data-guided conditions. We conclude our sample distribution covers the real distribution well.

Table 4: Estimated variance of our scores using bootstrap method.

(a) The ellipse data

Feature	Local score	Global score
Aspect ratio	$6.74e-4$	$3.40e-7$
Size	$9.85e-4$	$5.00e-7$

(b) The brain MRI ROI data

Feature	Local score	Global score
Hippo. vol.	$3.30e-3$	$2.75e-5$
Hippocampus	$3.20e-3$	$2.41e-5$

Given that the data distribution is well covered, we perform bootstrapping tests (Efron, 1992) to estimate score variation due to sampling. This involves resampling the existing samples 5,000 times, calculating a new score for each, and computing the variance of the resulting scores. The variances for local and global scores are reported in Table 4. For local scores, we compute the mean variance across feature values. The small variance suggests that our score remains informative even with relatively small sample sizes.

5. Conclusion and Discussion

Our proposed metrics are designed to assess the extent to which a classifier depends on a well-known, meaningful feature, either across the entire dataset or at specific feature value points. The results show that it effectively quantifies feature importance for classifiers, offering a normalized range for both local and global assessments. Furthermore, our metrics could be applied to raw imaging data, using foundational image generative models, as long as there is either a closed-form or deep learning-based mapping to the feature.

One limitation is the computational cost of generating samples with the diffusion model (30 hours on a single A100 GPU on brain ROI data), but this trade-off can be managed through evaluation, as shown in our experiments. Another limitation is that our metrics do not extend to the subject level, which remains a direction for future work.

Acknowledgments

This work was partially supported by NSF Smart and Connected Health grant 2205417.

References

Maximilian Augustin, Valentyn Boreiko, Francesco Croce, and Matthias Hein. Diffusion visual counterfactual explanations. *Advances in Neural Information Processing Systems*, 35:364–377, 2022.

Hyungjin Chung, Jeongsol Kim, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *The Eleventh International Conference on Learning Representations, ICLR 2023*. The International Conference on Learning Representations, 2023.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Bradley Efron. Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics: Methodology and distribution*, pages 569–593. Springer, 1992.

Bruce fischi, David H Salat, Evelina Busa, Marilyn Albert, Megan Dieterich, Christian Haselgrove, Andre Van Der Kouwe, Ron killiany, David kennedy, Shuna Klaveness, et al. Whole brain segmentation: Automated labeling of neuroanatomical structures in the human brain. *Neuron*, 33(3):341–355, 2002.

John Fox. *Applied regression analysis and generalized linear models*. Sage publications, 2015.

Yash Goyal, Amir Feder, Uri Shalit, and Been Kim. Explaining classifiers with causal concept effect (cace). *arXiv preprint arXiv:1907.07165*, 2019a.

Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. Counterfactual visual explanations. In *International Conference on Machine Learning*, pages 2376–2384. PMLR, 2019b.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Guillaume Jeanneret, Loïc Simon, and Frédéric Jurie. Diffusion models for counterfactual explanations. In *Proceedings of the Asian Conference on Computer Vision*, pages 858–876, 2022.

Yinzhu Jin. ellipse, 2022. URL https://pypi.org/project/ellipse/. Accessed: 2025-01-31.

Yinzhu Jin, Matthew B. Dwyer, and P. Thomas Fletcher. Measuring feature dependency of neural networks by collapsing feature dimensions in the data manifold. In *2024 IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 1–5, 2024. doi: 10.1109/ISBI56570.2024.10635874.

Diederik P Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Pamela J LaMontagne, Tammie LS Benzinger, John C Morris, Sarah Keefe, Russ Hornbeck, Chengjie Xiong, Elizabeth Grant, Jason Hassenstab, Krista Moulder, Andrei G Vlassenko, et al. Oasis-3: longitudinal neuroimaging, clinical, and cognitive dataset for normal aging and alzheimer disease. *medrxiv*, pages 2019–12, 2019.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. In *International Conference on Machine Learning*, pages 7176–7185. PMLR, 2020.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11): 559–572, 1901.

Walter HL Pinaya, Mark S Graham, Eric Kerfoot, Petru-Daniel Tudosiu, Jessica Dafflon, Virginia Fernandez, Pedro Sanchez, Julia Wolleb, Pedro F Da Costa, Ashay Patel, et al. Generative ai for medical imaging: extending the monai framework. *arXiv preprint arXiv:2307.15208*, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

Alessia Sarica, Roberta Vasta, Fabiana Novellino, Maria Grazia Vaccaro, Antonio Cerasa, Aldo Quattrone, and Alzheimer's Disease Neuroimaging Initiative. Mri asymmetry index of hippocampal subfields increases through the continuum from the mild cognitive impairment to the alzheimer's disease. *Frontiers in neuroscience*, 12:576, 2018.

Karl Schulz, Leon Sixt, Federico Tombari, and Tim Landgraf. Restricting the flow: Information bottlenecks for attribution. In *International Conference on Learning Representations*, 202.

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*, pages 618–626, 2017.

K Simonyan and A Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations (ICLR 2015)*. Computational and Biological Learning Society, 2015.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

Roman Solovyev, Alexandr A Kalinin, and Tatiana Gabruseva. 3d convolutional neural networks for stalled brain capillary detection. *Computers in Biology and Medicine*, 141: 105089, 2022. doi: 10.1016/j.compbiomed.2021.105089.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

Xinru Wang and Ming Yin. Are explanations helpful? a comparative study of the effects of explanations in ai-assisted decision-making. In *Proceedings of the 26th International Conference on Intelligent User Interfaces*, pages 318–328, 2021.

Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

Shen Zhu, Ifrah Zawar, Jaideep Kapur, and P Thomas Fletcher. Quantifying hippocampal shape asymmetry in alzheimer's disease using optimal shape correspondences. In *2024 IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 1–4. IEEE, 2024.

Appendix A. Experiments on CelebA dataset

In this section, we present experiments on the CelebA dataset with binary features, comparing our method to CaCE (Goyal et al., 2019a). This demonstrates the applicability of our approach to natural images, particularly when data is abundant and additional sampling is unnecessary.

A.1. Setup

CelebFaces Attributes (CelebA) (Liu et al., 2015) is a publicly available dataset of celebrity face photos annotated with multiple binary attributes. We cropped the images into squares and resized them to 128×128 pixels. We focus on the gender classification task. Among the binary annotations from the original dataset, we chose some of them that are apparently related or unrelated to gender as our target features (as listed in Table 5).

We trained a residual network (He et al., 2016) implementation by (Wightman, 2019) for the gender classification. Detailed architectural information for the specific variant we used can be found at `https://huggingface.co/timm/resnet10t.c3_in1k`. We opted not to use pre-trained weights as they did not improve classifier performance.Since the dataset is large enough and the feature is binary, it was not necessary to train a generative model.

A.2. Results

Table 5: Our proposed score and CaCE score for binary features on CelebA gender classifier.

Feature	Global (ours)	Local (ours)		CaCE
		negative	positive	
Wearing lipstick	0.639	0.266	0.982	−0.625
Heavy makeup	0.404	0.003	0.992	−0.771
Arched eyebrows	0.154	−0.067	0.708	−0.420
Beard	0.272	0.158	0.935	0.712
5 o'clock shadow	0.180	0.093	0.964	0.683
Blurry	$1e-4$	0.002	−0.027	0.029

The results of our proposed scores for various binary features in CelebA classification are presented in Table 5. Our analysis reveals that features related to makeup and facial hair are among the most significant, which aligns with real-world expectations. These features have scores significantly higher than the last feature, "blurry", which indicates photo blurriness and is unrelated to gender. Additionally, we observe that these important features exhibit much higher local scores in the positive class compared to the negative class. This indicates that while a face with makeup is strongly indicative of a female, a face without makeup could belong to either gender with considerable probabilities. The same interpretation applies to facial hair.

Our scores generally align with CaCE scores, with both methods assigning larger absolute values to important features. CaCE emphasizes the influence of feature values on predictions and indicates the direction of this influence: a negative value suggests a higher

likelihood of classification as female, while a positive value suggests a higher likelihood of classification as male. In contrast, our method evaluates the "usefulnes" of the target feature. For instance, two features related to facial hair receive relatively lower scores from our model due to the scarcity of positive samples in the dataset, which make up only 14.6% and 10.0%, respectively. Since these features are only strong indicators when being positive (as reflected by our local scores), assigning them lower importance is justified.

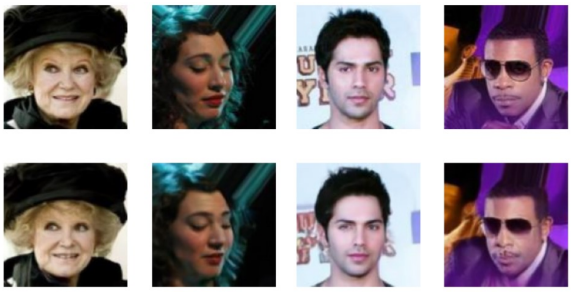


Figure 9: Original images (top row) from CelebA and corresponding counterfactual explanations (bottom row) generated using (Augustin et al., 2022).

When applying the diffusion CE, 62.2% of the generated explanation samples successfully flipped the model predictions. Sample pairs of original images and their corresponding counterfactual explanations, where the model's prediction was changed, are shown in Figure 9. The observed changes are generally quite subtle. This, along with the low rate of model prediction flipping, may be due to the classifier not trained for robustness. Additionally, the method appears to prioritize features affecting fewer pixels. For instance, it alters the eyebrow (shown in the first column) or lip makeup (shown in the third column) but not the facial hair. We believe this is related to the strategy to stay close to the original data point by minimizing the L_1 distance, and we anticipate similar results with L_2 distance. In contrast, we believe our method is not limited to robust classifiers and does not favor features affecting smaller regions.

Appendix B. Models

We denote the batch size as N.

B.1. Ellipse classifier and regression models

For the ellipse dataset, we used a basic convolutional neural network (CNN) (LeCun et al., 1998) classifier consisting of four convolutional layers and two linear layers since its simplicity. For the regression models on the aspect ratio and volume, we adopted the same architectures. The DDPM was trained with regular U-Net (Ronneberger et al., 2015) backbone.

The CNN block is a basic convolutional block as described in Table 7.

Table 6: Summary of the ellipse classifier and regression models architectures.

Layer Type	Output Size
Input	$(N, 1, 32, 32)$
CNNBlock	$(N, 32, 16, 16)$
CNNBlock	$(N, 64, 8, 8)$
CNNBlock	$(N, 64, 4, 4)$
CNNBlock	$(N, 64, 2, 2)$
Flatten	$(N, 256)$
Linear & ReLU	$(N, 64)$
Linear (& Sigmoid for the classifier)	$(N, 1)$

Table 7: CNN block used for the ellipse dataset.

Layer type	Kernel Size
Input	-
2D convolution	3×3
ReLU	-
2D max pooling	2×2

B.2. Brain ROI classifier and regression models

We employed a 3D variant of a residual network (Solovyev et al., 2022), designed to mimic the gender classifier we trained on CelebA, for AD classification and hippocampal volume regression. This version replaces 2D convolutions, poolings, and normalizations with their 3D counterparts, while keeping the kernel sizes, strides, number of channels, and batch normalization parameters unchanged. As before, we trained the model from scratch on our dataset.

B.3. U-Net used for the DDPM on 2D datasets

We employed a standard U-Net backbone for the DDPM trained on the ellipse and CelebA datasets. Although our metrics did not require a diffusion model for CelebA, we trained one to perform diffusion counterfactual explanation (CE). We used an implementation that is publicly available at https://github.com/lucidrains/denoising-diffusion-pytorch. The initial convolution dimensions were 32 for the ellipse data and 64 for the CelebA data. The down-sampling and up-sampling paths each consist of four blocks, with each block comprising two ResNet blocks and a linear attention module. The total number of learnable parameters is 9.2 M for the ellipse data and 35.7 M for CelebA. For further details, please refer to the aforementioned library.

B.4. The latent diffusion model on brain ROIs

Due to the high-dimensional nature of this data, we trained a latent diffusion model (Rombach et al., 2022), which combines a VAE and a diffusion model in the latent space.

We adopted the 3D variant from (Pinaya et al., 2023), publicly accessible at https://github.com/Project-MONAI/GenerativeModels. This library provides a 3D variant of the original latent diffusion model specifically designed for biomedical applications.

We used a shallow autoencoder that downsamples spatial dimensions by a factor of 2, yielding a latent dimension of $1 \times 32 \times 32 \times 32$. The encoder and decoder each include two ResNet blocks with internal channels of 32 and 64. The model has 1.2 M learnable parameters. For the U-Net, we used 3 blocks for both the downsampling and upsampling paths, with each block comprising two ResNet blocks. The internal channel numbers are 256, 512, and 768, respectively. The total number of trainable parameters is 424.3 M. For detailed information on the architectures, please refer to the aforementioned library.

Appendix C. Training setup

All classifiers and regression models are trained to optimize the performance on validation sets that are randomly split out from the training set. The performances on the test sets are shown in 8. The balanced accuracy is reported for AD prediction because of class imbalance in sample numbers. Due to the small sample size, each training sample from the brain ROI data was augmented with ten random 3D rotations with angles $\alpha \sim \mathcal{U}(0, 10°)$. The AD classifier was trained with a weighted sampler to counteract the unbalanced distribution.

Table 8: Performance of classifiers and regression models.

(a) Classifiers

Task	Accuracy
Gender classification on CelebA	0.966
Ellipse classification	0.906
AD prediction on brain ROI	0.836

(b) Regression models

Task	R^2 score
Ellipse aspect ratio prediction	0.999
Ellipse size prediction	0.998
Hippo. vol. prediction on brain ROI	0.864

C.1. Data preprocessing

Brain MRIs were cropped around the hippocampi and augmented with random 3D rotations up to 10 degrees. We used Freesurfer segmentations (fischi et al., 2002) from the original dataset to identify the hippocampus regions. We further adjusted the contrast by normalizing pixel intensities to the range 0 to 1 using the 10th and 90th percentiles as thresholds.

C.2. Classifiers and regression models

All classifiers and regression models were trained using the Adam optimizer with a learning rate of $1e-5$. An L_2 regularization with a weight of 1 was applied to the ellipse size regressor. Early stopping was used when performance on the validation set ceased to improve. However, none of the classifiers are specifically trained for robustness.

C.3. Diffusion models

For all diffusion models, we used a linear time schedule with $\beta_1 = 0.0015$ and $\beta_T = 0.0205$, and a total of $T = 1000$ time steps. The U-Net was trained to predict noise until full convergence on the training set, using the Adam optimizer with a learning rate of $1e-5$.

C.4. VAE for the latent diffusion

For the VAE used for dimensional reduction on the brain ROI data, we trained it with a combination of L_1 reconstruction loss, KL-divergence loss, and perceptual loss. The perceptual loss was computed using a SqueezeNet model trained on ImageNet, with 25% of 2D slices randomly selected along different dimensions. The KL-divergence term was weighted at $1e-7$ (summed across all latent dimensions), while the perceptual loss was weighted at $1e-3$.

During guided sampling, the gradient is backpropagated through the classifier and then through the decoder to the latent space. Since VAE decoder is trained with noise infusion, and the latent representation is decoded using this trained decoder, we assume this process enhances the robustness of the AD classifier guidance.

Appendix D. Diffusion sampling setup

For the noise coefficient, we used $\sigma_t = \sqrt{\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t}$ for all the DDPM sampling methods.

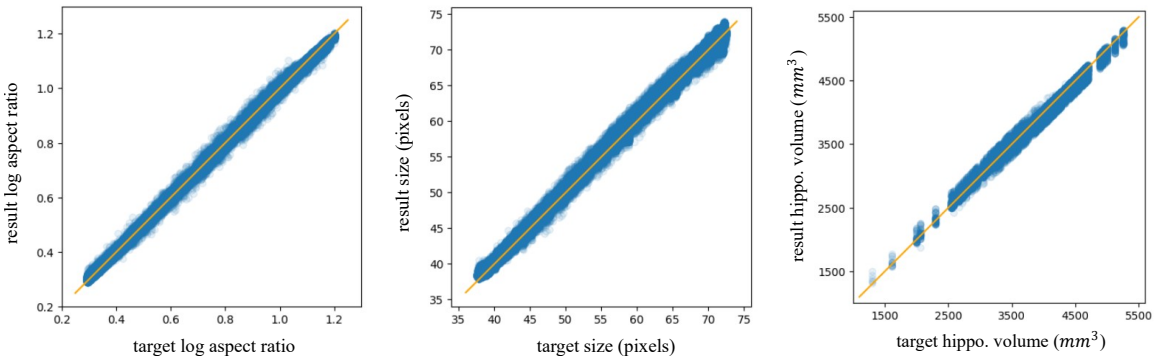
D.1. Constrained sampling (for our metrics)

We report the coefficient ρ (see Eq.(13)) we used for each feature in Table 9.

Table 9: ρ used for constrained sampling

Feature	ρ
Ellipse aspect ratio	0.1
Ellipse size	0.1
Hippocampal volume	0.03 ($t > 400$) 0.04 ($t \leq 400$)
Hippocampus	0.4

For the constrained hippocampus sampling, we dilated the ground truth hippocampi masks by 2 pixels to include edge information.



(a) Ellipse log aspect ratios (b) Ellipse sizes (c) Hippocampal volumes

Figure 10: Target and result feature values with constrained DDPM sampling.

The feature values were sampled from the real test data. We generated 200 samples per feature value and removed those with feature values (as evaluated by the regression model) deviated by more than ± 0.3 standard deviations from the constraint. This thresholding was not applied to the hippocampus feature in the brain ROI data due to the absence of a well-defined standard deviation. We present the result feature values corresponding to the target values in Figure 10, demonstrating that the feature values are well-constrained.

D.2. Diffusion counterfactual explanation

We used the same diffusion model as for the constrained sampling. The weights for the classifier and guidance, and the L_1 distance guidance to the original sample, are set to 0.1 and 0.15, respectively, as specified in their paper. We followed their paper (Augustin et al., 2022), starting sampling from the noisy images at time step $t = \frac{T}{2} = 500$.

Appendix E. Samples from the constrained sampling

Note that while our brain ROI data is 3D, we are only showing a central slice. As a result, it may not be intuitive to assess the constrained hippocampal volume from these 2D slices. For quantitative results, please refer to Figure 7(c).

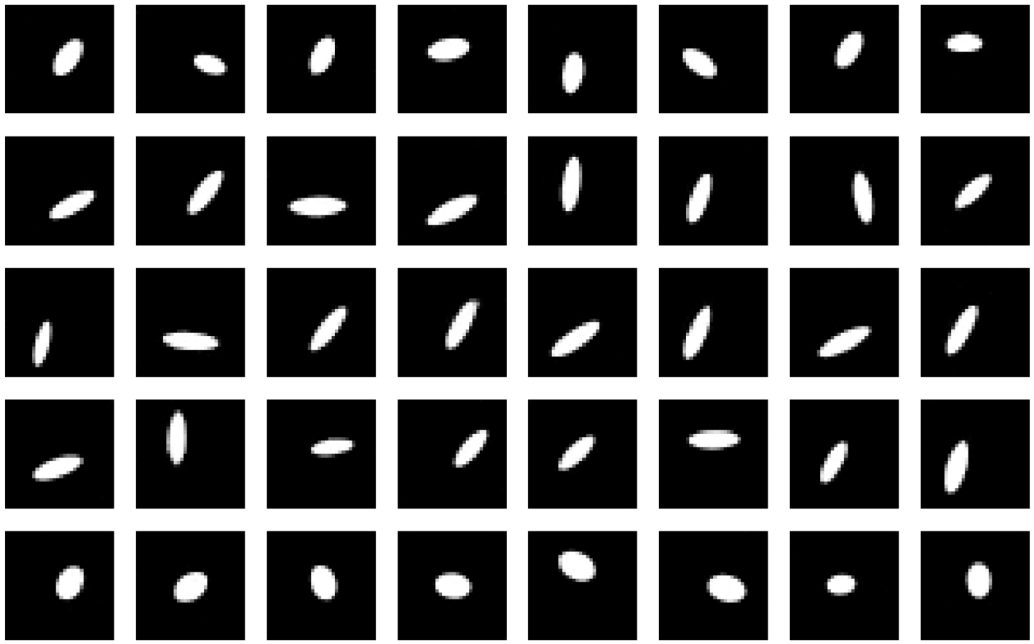


Figure 11: More samples with constrained aspect ratio: each row has the same target aspect ratio.

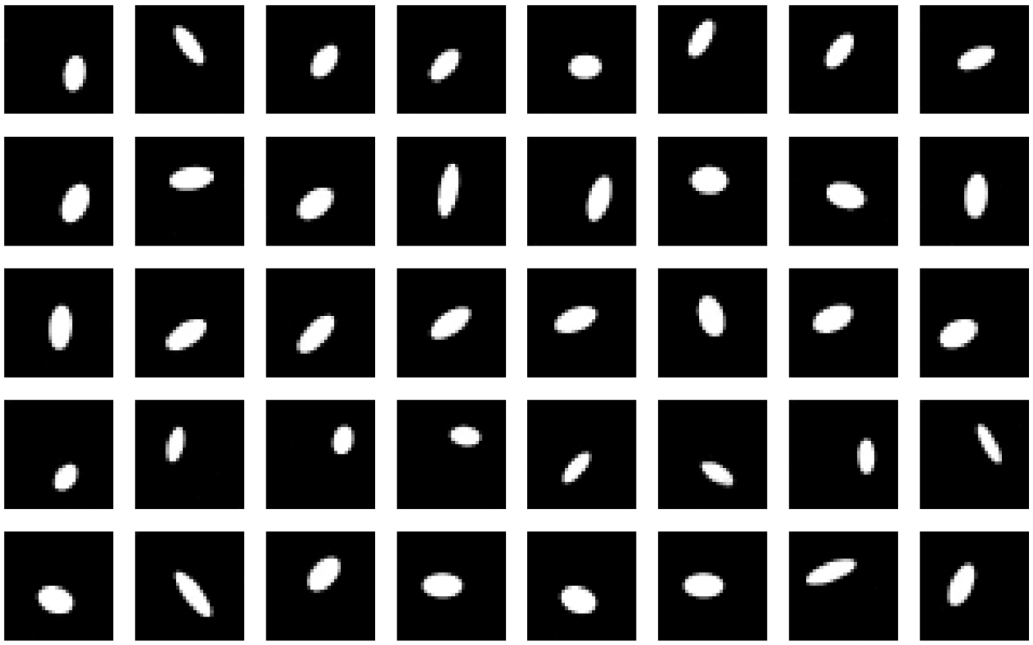


Figure 12: More samples with constrained size: each row has the same target size.

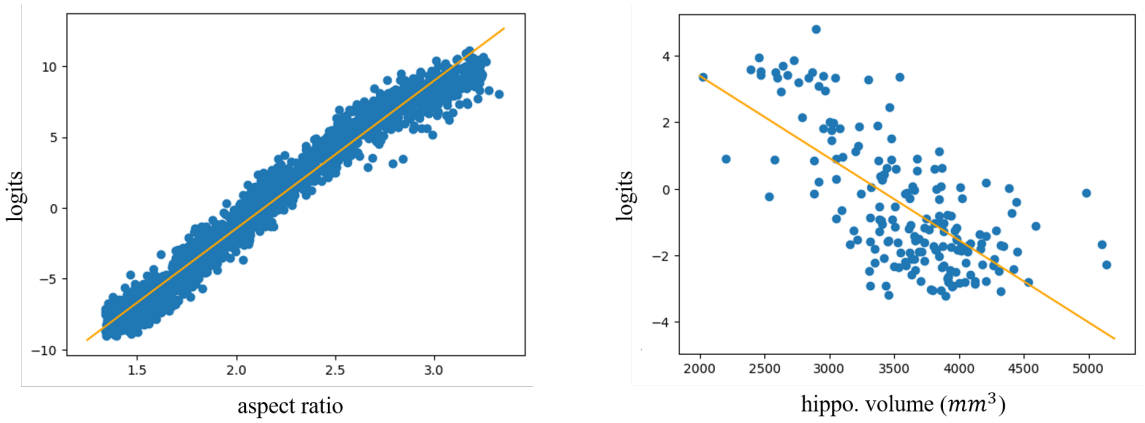
Figure 13: Samples with constrained hippocampal volume: each row has the same target volume.

Figure 14: Samples with constrained hippocampus: each row has the same target hippocampus.

Appendix F. Linear Regression Analysis

Given the analogy between our proposed metrics and the R^2 score used in linear regression, a natural question arises: could linear regression be applied to feature attribution? In Figure 15, we fit linear regression models to the classifier logits (the outputs before the Sigmoid activation), using the interpretable feature as the input. This allows us to calculate R^2 scores, which are 0.968 for the ellipse aspect ratio and 0.467 for hippocampal volume. However, these values are not directly comparable to our metrics, as they rely on continuous logits, whereas we use discrete classes. While this approach may seem reasonable for the given examples, we argue that there are several reasons why our method cannot be replaced by such a simple strategy:

- The method assumes a linear relationship between the interpretable feature and the classifier output, which is not always the case. The examples presented are not perfectly linear, and one could imagine a more extreme case where one class consists of ellipses with aspect ratios between 2 and 2.5, and the other class includes ellipses with aspect ratios either smaller than 2 or larger than 2.5. The resulting scatter plot would exhibit a U-shaped distribution.
- This method does not work well for more complex features like the hippocampal region, which lacks a fixed dimensionality in the raw input space.
- Unlike our method, this approach cannot produce local scores specific to given feature values. Additionally, it is insensitive to decision boundaries.



(a) Ellipse log aspect ratios (b) Hippocampal volumes

Figure 15: Scatter plots of feature values versus classifier logits (blue), with the corresponding fitted linear regression models (orange).